COMPUTE

the Club Of Microprocessor Programmers, Users, and Technical Experts

Georgia Marszalek, Editor • David Graves, Editor • Doug Hall, Technical Consultant

Sponsored by National Semiconductor Corp., Santa Clara, Ca. 95051 Vol. 3, No. 5, May 1977

Add a Programmable I/O Interface to your SC/MP Kit

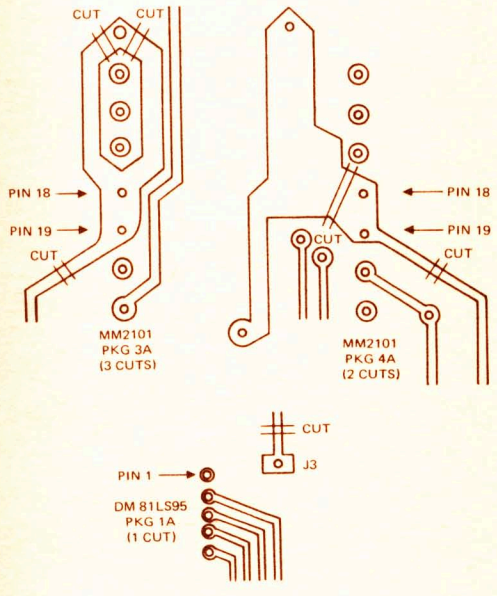
(submitted by Thomas M. Farr Jr., 569 Medina Dr., Highland Vill., TX 75067)

After toying around with my SC/MP kit for awhile, I began to use it for some serious investigation of real applications. Immediately, it became obvious that some I/O capability would be needed. The Intel 8255 I/O Port chip, now second-sourced by National (as INS8255), was a natural solution for this need.

The INS8255 may be programmed as three 8-bit unidirectional ports (either exclusively input or output); or as two 8-bit unidirectional ports with "handshaking"; or as one 8-bit bidirectional port with handshaking. Single bits in any port can be set or reset under software control. (For details, see the INS8255 Programmable Peripheral Interface data sheet included as the centerfold in this COMPUTE.)

To add the INS8255 to the SC/MP kit you must:

1. Make the cuts shown in figure 1 on the SC/MP kit P.C. card.



NOTE: THESE ARE THE SAME CUTS REQUIRED FOR THE "SCRIMP" MODIFICATIONS SHOWN IN "COMPUTE" VOL. 2, NO. 11, P. 6

Figure 1

(Continued on Page 18)

CRIMP!

by Tom Harper

Are you tired of waiting for your IMP16 cross assembler to punch SC/MP object tapes? Fatigued by sorting various levels of paper memories and feeding them into your LCDS? Irritated by having to switch your one TTY between LCDS and IMP16P because your boss is too cheap to buy another one? Read on friend — we offer you escape!

A peachy system has been set up in the Miami SC/MP school that shares one TTY with 2 (count them) prototyping systems. It also allows RLMs to be transferred directly from disc to LCDS memory. The software concept can be expanded to allow program tracing during execution. We are (trust us) working on it.

The LCDS system provides a resident debug utility which contains all of the subroutines necessary for loading and reading memory, initialization of resources and for transfer of control. These routines are normally invoked by TTY key depression, i.e.,

PRESS	SYSTEM WILL —
A	— call program to alter memory
then ØØ1	— set register to memory address 0001
,	— comma terminates entry of address
C4	— contents of location 0001 are set to ØC4
,	— comma terminates entry of data
ØØ	— etc.
,	— etc.
CR	— carriage return terminates operations

If the TTY output of any IMP16P or PACE prototyping system is connected to the TTY input of a Low Cost Development System it is not possible for the LCDS to detect that it is being controlled by a higher order system instead of a TTY/human operator combination. The high order system can now load the LCDS memory, as in the previous example, by generating the corresponding ASCII codes and presenting them to the TTY port. In a like manner the memory can be read. Any function available by TTY keyboard entry can now be called by a program resident in the higher order system. Elaborate operations can easily be performed by calling combinations of these subroutines in sequence.

This technique was successfully employed on the DEROACH system that was used for teaching SC/MP courses in the Eastern Education Center prior to the introduction of SC/MP-LCDS.

The program described here is an elaboration of DEROACH utilizing only standard hardware (except for the multiplexer) and requiring no wiring changes or other modifications. The program in its present form is called CRIMP.

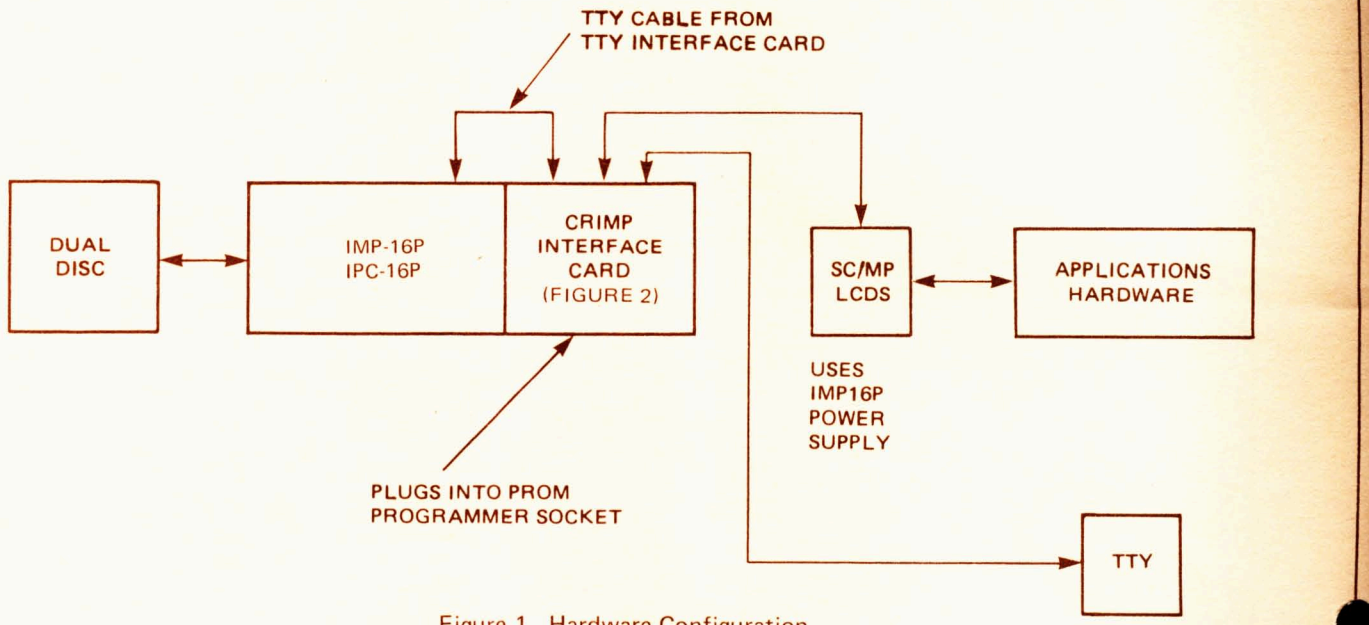


Figure 1. Hardware Configuration

Firmware Compatibility

Since CRIMP treats the LCDS debug utility as a bank of subroutines, no modification of that firmware is necessary.

All IMP16P/PACE-P systems are card reader compatible even though relatively few card readers are in field use. The utility described here can be coded into a pair of MM 5203Qs and used to replace the existing card reader driver PROMs. A G7FØØ command from any appropriate utility will force a jump to the CRIMP controller. The user may then work with his object applications program and when through return to a utility in order to modify or expand his source code.

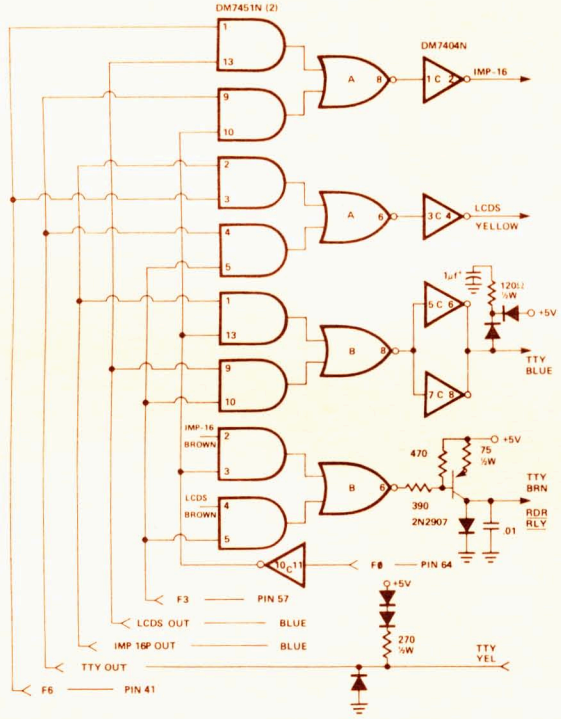


Figure 2

Circuitry

Figure 1 is a block diagram of the hardware configuration.

Figure 2 details the circuitry that is spliced into a standard TTY cable. It represents all that is required for the interface described. All non-TTY inputs and outputs are available at the prom programmer base connector of any NSC prototyping system.

The current loops are terminated only on the IMP16 TTY interface card. This will leave some wires open. Figure 3 shows the termination of unused wires.

System commands and operation are covered in detail in the listing.

FIGURE 3 – CABLE DETAIL

EDGE CONNECTOR ON IMP16P TTY BOARD			DESTINATION
PIN	COLOR	FUNCTION	
1	YELLOW	16P INPUT	PIN C2 MUX CARD
2	ORANGE	TERMINATION	TTY CABLE ORANGE
3	PURPLE	TERMINATION	TTY CABLE PURPLE
4	BLUE	16P OUT	PINS A2, B1 MUX
5	BROWN	READ SELECT	PIN B2 MUX CARD
6	GREEN	TERMINATION	TTY CABLE GREEN
TTY CABLE			
	YELLOW	TTY OUT	PINS A4, A9 MUX CARD
	ORANGE	TERMINATION	16P CABLE ORANGE
	PURPLE	TERMINATION	16P CABLE PURPLE
	BLUE	TTY INPUT	PINS C8, C6 MUX CARD
	BROWN	READER RELAY	TRANSISTOR COLLECTOR MUX CARD
	GREEN	TERMINATION	16P CABLE GREEN
LCDS TTY CABLE			
	YELLOW	LCDS INPUT	PIN C4 MUX CARD
9D10	ORANGE	ADXØ7200 DECODE	MUX CARD BASE PIN-120
	PURPLE	OPEN	
	BLUE	LCDS OUTPUT	PINS A13, B9 MUX CARD
	BROWN	READER RELAY	PIN B4 MUX CARD
	GREEN	OPEN	

```
                .TITLE  CRIMP,' IMP/LCDS INTERFACE'

                ; CRIMP COMMANDS
                ;
                ; THERE ARE TWO MODES OF OPERATION
                ; FOR THE CRIMP INTERFACE:
                ;
                ;       --->IMP<---
                ;
                ;           AND
                ;
                ;       --->LCDS<---
                ;
                ; IN --->IMP<--- MODE THERE ARE 4
                ; OPERATIONS AVAILABLE:
                ;
                ; L - DEPRESSION OF THE L KEY SWITCHES
                ; THE TTY TO THE SC/MP LCDS. THE
                ; LCDS MUST BE INITIALIZED PRIOR TO
                ; ISSUING THE L COMMAND. THE TTY WILL
                ; RESPOND WITH LCDS ?- TO INDICATE
                ; THAT LCDS MODE HAS BEEN ENTERED.
                ;
                ; D - THE D COMMAND LOADS ONE 8 BIT RLM
                ; FROM DISC AND TRANSFERS IT TO THE
                ; APPROPRIATE MEMORY LOCATIONS IN LCDS
                ; READ/WRITE MEMORY. THE TRANSFER RATE
                ; FOR THE DATA IS ONLY SLIGHTLY FASTER
                ; THAN THAT REQUIRED TO LOAD A PAPER TAPE.
                ; THE COMMAND FORMAT IS:
                ; D(FIRST RLM SECTOR)/(LAST RLM SECTOR) (CR)
                ;
                ;                   I.E.
                ;
                ; D 0186/0186 (CR) WILL LOAD AN 8 BIT
                ; RLM COMPLETELY CONTAINED IN DISC SECTOR
                ; 0186.
                ;
                ; D 0180/018F (CR)WILL LOAD AN 8 BIT RLM
                ; COMPLETELY CONTAINED IN DISC SECTORS
                ; 0180 THROUGH 018F.
                ;
                ; G - THE G COMMAND ALLOWS CONTROL TO BE
                ; TRANSFERED TO A PROGRAM STORED ON DISC
                ; AND BEGINNING AT THE SECTOR SPECIFIED
                ; IN THE COMMAND.
                ;
                ;               I.E.
                ;
                ; G 05C (CR) WILL LOAD AND TRANSFER CONTROL
                ; TO EDIT 16.
                ;
                ; P - THE P COMMAND FORCES A JUMP TO
                ; LOCATION 0FFFE. THIS ALLOWS THE USER
                ; TO ENTER A FIRMWARE UTILITY LOCATED ON
                ; THE IMP16C CPU CARD. AT THE MIAMI
                ; EDUCATION CENTER ALL IMP SYSTEMS HAVE A
                ; DEBUG UTILITY ON THE CPU CARD.
                ; EGOCENTRISM OBVIATES THE ATTITUDE THAT
                ; EVERYONE ELSE IN THE WORLD HAS FOLLOWED
                ; SUIT. THOSE WHO HAVE NOT MAY OBTAIN THE
                ; PROGRAM FROM THE USERS GROUP.
                ;
                ; IN --->LCDS<--- MODE ALL NORMAL
                ; LCDS DEBUG COMMANDS ARE FUNCTIONAL.
                ; ONE ADDITIONAL COMMAND HAS BEEN
                ; IMPLIMENTED TO RETURN CONTROL TO
                ; IMP. THE COMMAND G7200 SWITCHES
                ; THE TTY BACK TO THE IMP 16.
                ;
                ; THE CRIMP PROGRAM CAN BE LOADED FROM
                ; DISC OR MADE RESIDENT AS PROMS REPLACING
                ; THE EXISTING CARD READER FIRMWARE. THE
                ; PROM POSITIONS ON THE TTY CARD ARE:
                ;
                ;    ****************************************
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    *          HI          LO              *
                ;    *          ORDER       ORDER           *
                ;    *  ******  ******  ******  ******      *
                ;    *  *    *  *    *  *    *  *    *      *
                ;    *  *TTY *  * CR *  *TTY *  * CR *      *
                ;    *  *    *  *    *  *    *  *    *      *
                ;    *  ******  ******  ******  ******      *
                ;    *                                      *
                ;    *                                      *
                ;    *                                      *
                ;    ****************************************
                ;
                       .PAGE
                ;
                ; -----> THE HANDY TTY COMMAND PASTY <-----
                ;
                ;      ******************************
                ;      * L - TRANSFERS CONTROL TO    *
                ;      * LCDS DEBUG                  *
                ;      *                            *
                ;      * DXXXX/XXXX-LOADS            *
                ;      *   8 BIT RLM FROM           *
                ;      *   DISC AND TRANSFERS       *
                ;      *   TO LCDS R/W MEMORY       *
                ;      *                            *
                ;      * GXXXX-LOADS AND             *
                ;      *   EXECUTES PROGRAM         *
                ;      *   FROM DISC.               *
                ;      *                            *
                ;      * P-JUMPS TO DEBUG P          *
                ;      *                            *
                ;      * G7200-TRANSFERS             *
                ;      *   CONTROL FROM LCDS        *
                ;      *   DEBUG TO CRIMP           *
                ;      *                            *
                ;      ******************************
                ;
                ;      THE CRIMP ENTRY ADDRESS IS 07F00
                ;
                ;      WHEN USING THIS INTERFACE
                ;      SYSTEM POWER FOR THE LCDS
                ;      SHOULD BE DERIVED FROM THE
                ;      IMP 16P NOT AN EXTERNAL
                ;      POWER SUPPLY!!!!!!!!!!!!
                ;
                ;      IF THIS IS NOT OBSERVED
                ;      CRITICAL GROUNDING PROBLEMS
                ;      MAY NEGATIVELY INFLUENCE
                ;      THE DISC WRITE AMPLIFIER!!!!

                       .PAGE
                       .ASECT
                       .=07F00
                       .LOCAL

7F00 0A80 A ENTER:  PFLG   2,0          ;AVOID TROUBLE
7F01 2D2E A         JSR    @DIR         ;IDENTIFY PROGRAM
7F02 7FBC A         .WORD  ID
7F03 4C3E A PRMPT:  LI     0,'>'/256    ;PROMPT
7F04 2D2E A         JSR    @PUTC
7F05 2932 A GET:    JSR    GET1         ;GET FIRST COMMAND
7F06 21FC A         JMP    PRMPT        ;ABORT COMMAND
7F07 F120 A SCAN:   SKNE   0,NGS
7F08 2108 A         JMP    GC           ;IDENTIFY COMMAND
7F09 F11F A         SKNE   0,IN         ;GROUP
7F0A 2108 A         JMP    DC
7F0B F11E A         SKNE   0,LIFE
7F0C 2109 A         JMP    LCDT
7F0D F119 A         SKNE   0,THI
7F0E 2517 A         JMP    @BEST
7F0F 2930 A         JSR    ERROR        ;NONE OF THE ABOVE
7F10 21F4 A         JMP    GET          ;-ONE MORE TIME-
7F11 2D21 A GC:     JSR    @PUTC        ;ECHO 'G'
7F12 210D A         JMP    GAD
7F13 2D1F A DC:     JSR    @PUTC        ;DISC OPERATION
7F14 212E A         JMP    DSKIN        ;DISC--->LCDS
7F15 292A A         JSR    ERROR        ;DOES NOT COMPUTE
7F16 2D1C A LCDT:   JSR    @PUTC
7F17 2D18 A         JSR    @DIR
7F18 7FC0 A         .WORD  LDS
7F19 2D0B A         JSR    @THE
7F1A 4C0D A         LI     0,0D
7F1B 2D17 A         JSR    @PUTC
7F1C 2101 A         JMP    .+2
7F1D 7FC8 A         .WORD  LCDS
7F1E 2DFE A         JSR    @.-1
7F1F 21E0 A         JMP    ENTER
7F20 2978 A GAD:    JSR    HXFCH
7F21 21DE A         JMP    ENTER
7F22 3481 A         RCPY   1,0
7F23 2500 A         JMP    @.+1
7F24 C000 A         .WORD  0C000
7F25 7FC4 A THE:    .WORD  BKTBK
7F26 FFFE A BEST:   .WORD  0FFFE
7F27 0050 A THI:    .WORD  'P'/256
7F28 0047 A NGS:    .WORD  'G'/256
7F29 0044 A IN:     .WORD  'D'/256
7F2A 004C A LIFE:   .WORD  'L'/256
7F2B 000D A ARE:    .WORD  0D
7F2C 007F A GEN:    .WORD  07F
7F2D 007D A ER:     .WORD  07D
7F2E 0D0A A ALLY:   .WORD  0D0A
7F2F 7E73 A VERY:   .WORD  07E73
7F30 7EC3 A DIR:    .WORD  07EC3
7F31 7ED3 A TY:     .WORD  07ED3
     0007 A BELL    =      07
7F32 7E3B A GETC:   .WORD  07E3B
7F33 7E59 A PUTC:   .WORD  07E59
7F34 C008 A DISKIO: .WORD  0C008
7F35 1F00 A PLIST:  .WORD  01F00
7F36 2000 A BUFADR: .WORD  02000
7F37 0100 A BUFLNT: .WORD  256

7F38 2DF9 A GET1:   JSR    @GETC        ;GET CHARACTER
7F39 61F2 A         AND    0,GEN
7F3A F1F2 A         SKNE   0,ER         ;ABORT?
7F3B 2101 A         JMP    .+2          ;YES!!
7F3C 0201 A         RTS    1
7F3D 81F0 A         LD     0,ALLY       ;CARRIAGE RETURN
7F3E 2DF2 A         JSR    @TY          ;LINE FEED
7F3F 0200 A         RTS                 ;AND RETURN

7F40 4C07 A ERROR:  LI     0,BELL
7F41 2DF1 A         JSR    @PUTC
7F42 0200 A         RTS

7F43 4C02 A DSKIN:  LI     0,2          ;SET READ DISC MODE
7F44 8DF0 A         LD     3,PLIST      ;PARAMETER LIST ADDR
7F45 A300 A         ST     0,(3)
7F46 81EF A         LD     0,BUFADR     ;SET UP PLIST
7F47 A302 A         ST     0,2(3)
7F48 2950 A         JSR    HXFCH        ;GET FIRST SECTOR
7F49 21B9 A         JMP    PRMPT
7F4A 2DE8 A         JSR    @PUTC
7F4B A701 A         ST     1,1(3)       ;STORE LOGICAL SECTOR
7F4C 294C A         JSR    HXFCH        ;GET SECOND SECTOR
7F4D 21B5 A         JMP    PRMPT
7F4E 3801 A         NOP
7F4F 3801 A         NOP                 ;CANNOT BE 0
7F50 A706 A         ST     1,6(3)       ;STORE FINAL SECTOR
```

```
225 7F51 2DE2 A READ:   JSR     @DISKIO     ;READ ONE SECTOR
226 7F52 1F00 A         .WORD   01F00
227 7F53 3801 A         NOP
228 7F54 8301 A         LD      0,1(3)      ;LOAD THIS SECTOR #
229 7F55 F306 A         SKNE    0,6(3)      ;FINAL SECTOR?
230 7F56 2136 A         JMP     XFER        ;YES-
231 7F57 8302 A         LD      0,2(3)
232 7F58 C1DE A         ADD     0,BUFLNT    ;NO-
233 7F59 A302 A         ST      0,2(3)
234 7F5A 7B01 A         ISZ     1(3)        ;INCREMENT SECTOR #
235 7F5B 21F5 A         JMP     READ        ;DO IT AGAIN
236 7F5C 21E3 A         JMP     ERROR
237
238 7F5D 4C0D A LDLC:   .WORD   04C0D,0
    7F5E 0000 A
239 7F5F 7E59 A $PUTC:  .WORD   07E59
240
241 7F60 8DD5 A DMSG:   LD      3,BUFADR
242 7F61 8300 A RKD:    LD      0,(3)
243 7F62 A092 A         ST      0,TEMP1
244 7F63 6115 A         AND     0,STXM
245 7F64 F112 A         SKNE    0,START
246 7F65 2114 A         JMP     CRD
247 7F66 F10F A         SKNE    0,DATA
248 7F67 2112 A         JMP     CRD
249 7F68 F110 A         SKNE    0,STXM
250 7F69 2106 A         JMP     TERM
251 7F6A 0200 A         RTS
252
253 7F6B 810C A CALC:   LD      0,LMSK
254 7F6C 6092 A         AND     0,TEMP1
255 7F6D 4802 A         AISZ    0,2
256 7F6E A092 A         ST      0,TEMP1
257 7F6F 0200 A         RTS
258
259 7F70 29FA A TERM:   JSR     CALC
260 7F71 4C02 A         LI      0,2
261 7F72 2DEC A         JSR     @$PUTC
262 7F73 2912 A         JSR     DRK
263 7F74 290B A         JSR     KRLF
264 7F75 211B A         JMP     FINIS
265
266 7F76 0040 A DATA:   .WORD   040
267 7F77 0000 A START:  .WORD   000
268 7F78 003F A LMSK:   .WORD   03F
269 7F79 00C0 A STXM:   .WORD   0C0
270      0092 A TEMP1   =       092
271
272 7F7A 29F0 A CRD:    JSR     CALC
273 7F7B 4C02 A         LI      0,2
274 7F7C 2DE2 A         JSR     @$PUTC
275 7F7D 2908 A         JSR     DRK
276 7F7E 2901 A         JSR     KRLF
277 7F7F 21E1 A         JMP     RKD
278 7F80 4C0D A KRLF:   LI      0,0D
279 7F81 2DDD A         JSR     @$PUTC
280 7F82 4C0A A         LI      0,0A
281 7F83 2DDB A         JSR     @$PUTC
282 7F84 0200 A         RTS
283 7F85 4B01 A         AISZ    3,1
284 7F86 8300 A DRK:    LD      0,(3)
285 7F87 2DD7 A         JSR     @$PUTC
286 7F88 7C92 A         DSZ     TEMP1
287 7F89 21FB A         JMP     DRK-1
288 7F8A 4B01 A         AISZ    3,1
289 7F8B 0200 A         RTS
290
291 7F8C 7FD4 A         .WORD   CMSG
292
293 7F8D 2DFE A XFER:   JSR     @.-1
294 7F8E 7F5D A         .WORD   LDLC
295 7F8F 2934 A         JSR     BKTBK
296 7F90 21CF A         JMP     DMSG
297 7F91 293D A FINIS:  JSR     IMP
298 7F92 2500 A         JMP     @.+1
299 7F93 7F00 A         .WORD   ENTER
300 7F94 0046 A FTSX:   .WORD   046
301 7F95 002F A TWOF:   .WORD   02F
302 7F96 0039 A TTN:    .WORD   039
303 7F97 002F A DIA:    .WORD   '/'/256
304      0090 A SIZE    =       090
305 7F98 7ED3 A $PUT2C: .WORD   07ED3
306
307 7F99 4D00 A HXFCH:  LI      1,0         ;CLEAR AC1
308 7F9A 299D A         JSR     GET1        ;GET ONE CHARACTER
309 7F9B 0200 A         RTS                 ;ALT MODE RETURN
310 7F9C 3281 A         RCPY    0,2         ;TEMPORARY SAVE
311 7F9D F18D A         SKNE    0,ARE       ;TERMINATE
312 7F9E 2106 A         JMP     RETURN      ;YES-RETURN
313 7F9F F1F7 A         SKNE    0,DIA
314 7FA0 0201 A         RTS     1
315 7FA1 2906 A         JSR     HEXCN       ;NO-CONVERT TO BCD
316 7FA2 5D04 A         SHL     1,4         ;MOVE OVER
317 7FA3 3182 A         RXOR    0,1         ;ENTER HEX CHARACTER
318 7FA4 21F5 A         JMP     HXFCH+1     ;GET ANOTHER
319 7FA5 8188 A RETURN: LD      0,ALLY      ;CARRIAGE RETURN
320 7FA6 2DF1 A         JSR     @$PUT2C     ;LINE FEED AND
321 7FA7 0201 A         RTS     1           ;GO BACK
322
323 7FA8 E1EC A HEXCN:  SKG     0,TWOF      ;#?
324 7FA9 2106 A         JMP     BAD
325 7FAA E1EB A         SKG     0,TTN       ;LETTER?
326 7FAB 2108 A         JMP     NMBR
327 7FAC E1C9 A         SKG     0,DATA
328 7FAD 2102 A         JMP     BAD
329 7FAE E1E5 A         SKG     0,FTSX
330 7FAF 2106 A         JMP     HEX
331 7FB0 298F A BAD:    JSR     ERROR
332 7FB1 21E8 A         JMP     HXFCH+1
333
334 7FB2 7E59 A PUTK:   .WORD   07E59
```

```
335 7FB3 000F A NMSK:   .WORD   0F
336
337 7FB4 61FE A NMBR:   AND     0,NMSK
338 7FB5 2102 A         JMP     ECO
339 7FB6 61FC A HEX:    AND     0,NMSK
340 7FB7 4809 A         AISZ    0,9
341 7FB8 3880 A ECO:    RXCH    2,0
342 7FB9 2DF8 A         JSR     @PUTK
343 7FBA 3880 A         RXCH    2,0
344 7FBB 0200 A         RTS
345
346 7FBC 494D A ID:     .ASCII  'IMP'
    7FBD 5020 A
347 7FBE 0D0A A         .WORD   0D0A,0
    7FBF 0000 A
348 7FC0 4C43 A LDS:    .ASCII  'LCDS'
    7FC1 4453 A
349 7FC2 0D0A A         .WORD   0D0A,0
    7FC3 0000 A
350
351                     .LOCAL
352 7FC4 0800 A BKTBK:  SFLG    0,0
353 7FC5 0B80 A         PFLG    3,0
354 7FC6 0E00 A         SFLG    6,0
355 7FC7 0200 A         RTS
356
357 7FC8 0800 A LCDS:   SFLG    0,0
358 7FC9 0E80 A         PFLG    6,0
359 7FCA 0B00 A         SFLG    3,0
360 7FCB 1FFF A TEST:   BOC     15,.
361 7FCC 2102 A         JMP     IMP
362
363 7FCD 7EC3 A $MSG:   .WORD   07EC3
364 7FCE 7E59 A $PUTC:  .WORD   07E59
365
366 7FCF 0B80 A IMP:    PFLG    3,0
367 7FD0 0E80 A         PFLG    6,0
368 7FD1 0880 A         PFLG    0,0
369 7FD2 0200 A         RTS
370
371      0080 A BEGIN   =       080
372      0081 A END     =       081
373      0082 A MEMB    =       082
374      0083 A MEME    =       083
375      0084 A STBF    =       084
376      0094 A NBLS    =       094
377      0095 A PAIRS   =       095
378      0096 A $T      =       096
379
380 7FD3 0201 A OUT:    RTS     1
381
382 7FD4 4700 A CMSG:   PULL    3
383 7FD5 4300 A         PUSH    3
384 7FD6 8F00 A         LD      3,(3)
385 7FD7 29EC A         JSR     BKTBK
386 7FD8 8300 A $LOAD:  LD      0,(3)
387 7FD9 11F9 A         BOC     1,OUT
388 7FDA 5808 A         ROL     0,8
389 7FDB 2DF2 A         JSR     @$PUTC
390 7FDC 5C78 A         SHR     0,8
391 7FDD 11F5 A         BOC     1,OUT
392 7FDE 2DEF A         JSR     @$PUTC
393 7FDF 4B01 A         AISZ    3,1
394 7FE0 21F7 A         JMP     $LOAD
395      7F00 A         .END    ENTER
```

```
ALLY   7F2E  A     ARE    7F2B  A
BAD    7FB0  A     BEGIN  0080  A*
BELL   0007  A     BEST   7F26  A
BKTBK  7FC4  A     BUFADR 7F36  A
BUFLNT 7F37  A     CALC   7F6B  A
CMSG   7FD4  A     CRD    7F7A  A
DATA   7F76  A     DC     7F13  A
DIA    7F97  A     DIR    7F30  A
DISKIO 7F34  A     DMSG   7F60  A
DRK    7F86  A     DSKIN  7F43  A
ECO    7FB8  A     END    0081  A*
ENTER  7F00  A     ER     7F2D  A
ERROR  7F40  A     FINIS  7F91  A
FTSX   7F94  A     GAD    7F20  A
GC     7F11  A     GEN    7F2C  A
GET    7F05  A     GET1   7F38  A
GETC   7F32  A     HEX    7FB6  A
HEXCN  7FA8  A     HXFCH  7F99  A
ID     7FBC  A     IMP    7FCF  A
IN     7F29  A     KRLF   7F80  A
LCDS   7FC8  A     LCDT   7F16  A
LDLC   7F5D  A     LDS    7FC0  A
LIFE   7F2A  A     LMSK   7F78  A
MEMB   0082  A*    MEME   0083  A*
NBLS   0094  A*    NGS    7F28  A
NMBR   7FB4  A     NMSK   7FB3  A
OUT    7FD3  A     PAIRS  0095  A*
PLIST  7F35  A     PRMPT  7F03  A
PUTC   7F33  A     PUTK   7FB2  A
READ   7F51  A     RETURN 7FA5  A
RKD    7F61  A     SCAN   7F07  A*
SIZE   0090  A*    START  7F77  A
STBF   0084  A*    STXM   7F79  A
TEMP1  0092  A     TERM   7F70  A
TEST   7FCB  A*    THE    7F25  A
THI    7F27  A     TTN    7F96  A
TWOF   7F95  A     TY     7F31  A
VERY   7F2F  A*    XFER   7F8D  A
$LOAD  7FD8  A     $MSG   7FCD  A*
$PUT2  7F98  A     $PUTC  7F5F  A
$PUTC  7FCE  A     $T     0096  A*

NO ERROR LINES
SOURCE CHECKSUM =84AC
```

Software setup eases traffic flow for multiprocessors

by Janak Pathak,
National Semiconductor Corp., Santa Clara, Calif

Reprinted from Electronics, March 31, 1977; copyright © McGraw-Hill, Inc., 1977.

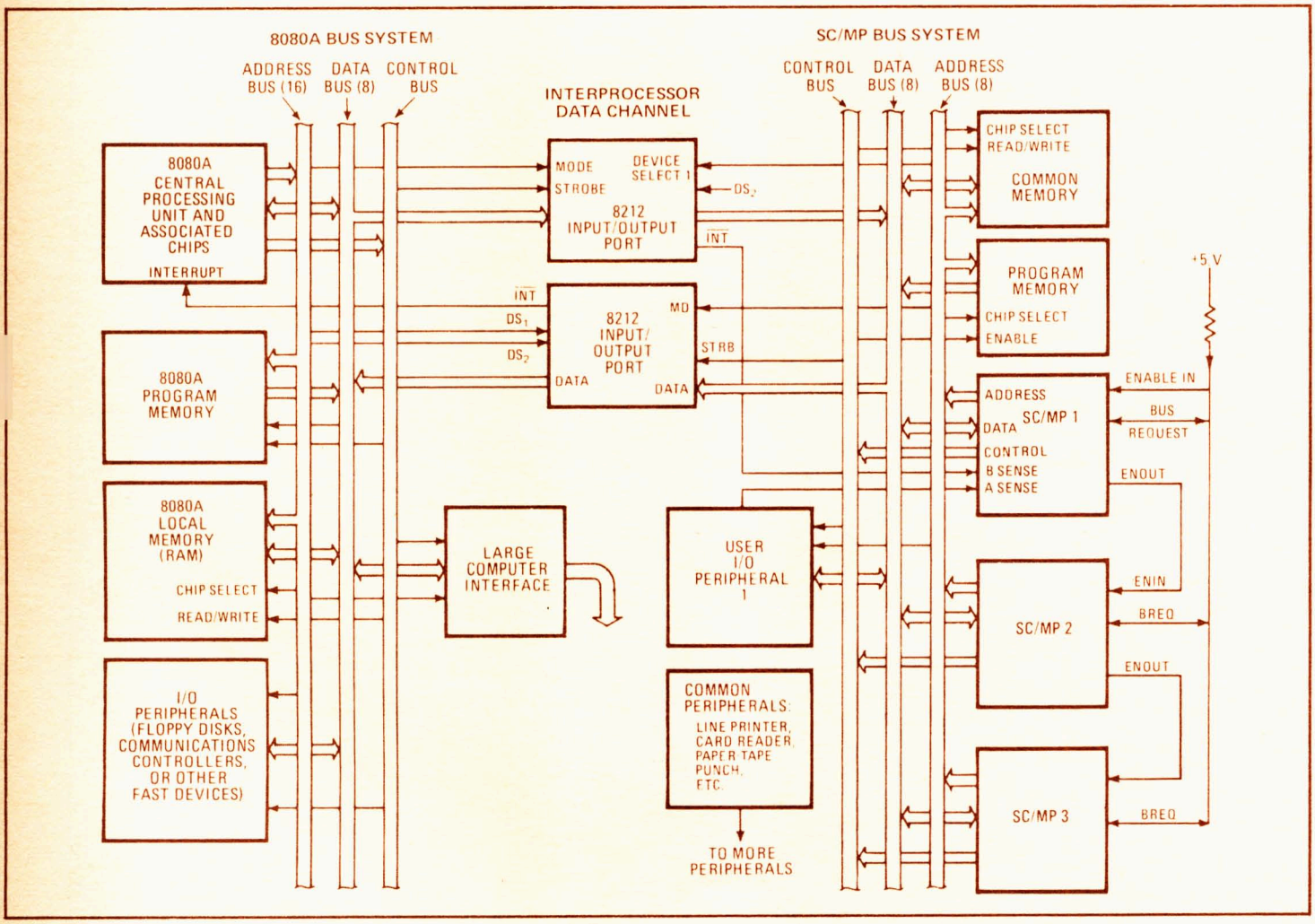
☐ Setting up the software that turns a melange of slow, serial microprocessors into a speedy multiprocessor hierarchy is largely a matter of assuring that the traffic flow among the devices and between them and the outside world is smooth, orderly, and quick. Once the designer has chosen a good device combination for the system—such as an 8080A and several SC/MPs—he faces the problem of transferring data and instructions between the processors. A related and nearly as important a problem is communication with the system's operator through an input/output terminal.

By allowing one of the SC/MPs to perform these supervisory jobs, the 8080A and the other SC/MPs are free to perform the tasks assigned to the system. The 8080A has a powerful instruction set and relatively fast execution time, which are suited to interfacing with high-speed peripherals, while the inexpensive SC/MPs' simplicity makes them an excellent choice for the job of handling the interfaces to less complex, slower peripheral equipment.

In a typical system, three or more SC/MPs (one controlling interprocessor information transfers) can control the input/output operations of such low-speed peripherals as a teletypewriter, a cathode-ray-tube display, a line printer, and such low-speed interfaces as analog-to-digital converters and sensors. The system's 8080A controls high-speed peripherals such as floppy disks and communications controllers.

There are three basic configurations for the multiprocessor system:

- All SC/MPs perform different tasks under the control of the 8080A, which also executes its own program.
- All SC/MPs perform the same tasks under the control of the 8080A, which again executes its own program.
- All SC/MPs perform their own tasks and use the 8080A either as a data reference or as a higher-level processor.



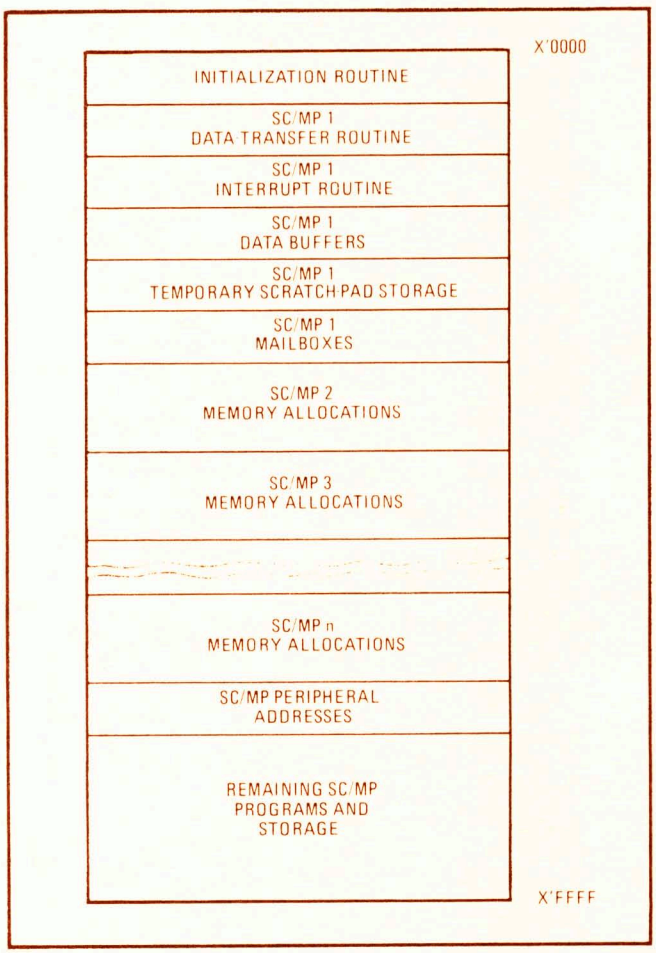
1. Multiprocessor. One 8080 and three SC/MPs are combined, each type of device handling jobs appropriate to its capability. SC/MP 1 supervises transfer of data between the other processors, as well as serving an interface with the operator's I/O device.

Who does what

In the first two setups, the supporting SC/MPs execute macroinstructions from the 8080A, while it monitors them and processes the assembled data. Both of these are "master/slave" arrangements, with the more powerful 8080A serving as master. The third setup is an example of the other possibility, a "master/master" arrangement. With any of the configurations, the 8080A may also function as a slave for a remotely located, large computer.

A bidirectional interface (Fig. 1) allows data transfers between the SC/MP and 8080A microprocessors. The controlling SC/MP (No. 1) transfers the interprocessor data using interrupts. The other SC/MPs execute independent programs while sharing a common memory with the controlling SC/MP.

The system includes individual address-, data-, and control-bus systems for each microprocessor. In the 8080A portion, the program read-only memory, local random-access memory, and peripherals are connected to the central processing unit and its accessory chips through the associated bus system. In the other portion, the program ROM, common RAM, and peripherals are connected to the SC/MP through the bus system. For this portion of the system, the designer must assign locations in the common memory for the data buffers, mailboxes, and temporary (scratch-pad) storage for each SC/MP (Fig. 2).



2. Memory organization. A common memory is used for all SC/MPs in the system. When a SC/MP requires service, it places a request in the mailbox, which is polled by the controlling device (which takes the most of the memory, since it has supervisory tasks.)

BIT NUMBER							
7	6	5	4	3	2	1	0
MODE		DESTINATION		SOURCE		OPERATION	
READ = 00		8080 = 00		8080 = 00		START = 00	
WRITE = 01		SC/MP 1 = 01		SC/MP 1 = 01		TRANSFER = 01	
STATUS = 10		SC/MP 2 = 10		SC/MP 2 = 10		ABORT = 10	
N. U. = 11		SC/MP 3 = 11		SC/MP 3 = 11		TERMINATE = 11	

3. Selection character. An 8-bit selection character is used by the 8080 and the SC/MPs to designate the type of data transfer. The 8080 places its selection character in the 8212 I/O port buffer, while each SC/MP places its selection character in a mailbox in memory.

Since each SC/MP treats the peripheral and memory devices identically, all of them can share the peripherals connected to the bus system. However, SC/MP 1 is assigned the tasks of transferring data between the microprocessors and of communicating with the operator's primary input/output device (user I/O peripheral 1 in Fig. 1). The assignment of the transfer task to one SC/MP eliminates any waiting period from the difference in operating speeds between the SC/MP and 8080A. It also relieves the 8080A from performing a task that does not require its sophistication.

When a particular microprocessor—either the 8080A or a SC/MP—must initiate a transfer, it does so with an 8-bit word called a selection character (Fig. 3). The 8080A places its selection character in one of the 8212 I/O port buffers, which will set the interrupt on the sense B line of the controlling SC/MP. Each SC/MP places its selection character in an assigned memory location (a mailbox). If the controlling SC/MP detects a selection character while polling the microprocessors for a service request, the requesting device may then initiate an information transfer.

The bus-control logic allows individual SC/MP access to the bus for data transfers and permits bus access by priority, thus achieving maximum bus utilization. During execution, some instructions need only one access to the bus. Others need two accesses, and a few need three. So the bus is idle during most of the processing time, permitting other processors to use it to execute their programs.

Obtaining bus access

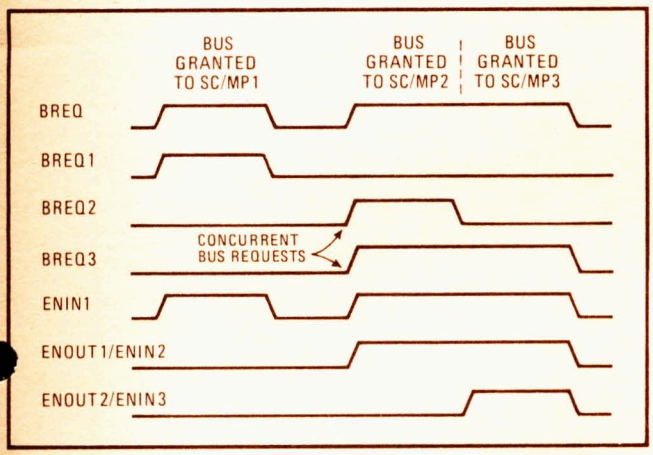
Requesting and gaining bus access and resolving priorities in the event of concurrent bus requests takes three signals: the bus request, BREQ, the enable input, ENIN, and the enable output, ENOUT.

The BREQ signal is bidirectional and is wire-ORed to the bus-request terminal of each SC/MP in the system. Any SC/MP wishing access to the bus must wait for a low BREQ, indicating the common bus-request line is free. Then the processor may activate its bus-request line.

The ENIN signal grants the processor access to the bus. It is driven high to initiate a data transfer. At the completion of the data transfer, the bus request is deacti-

vated, and the ENOUT signal is activated. It indicates that the processor has freed the buses.

For bus coordination and for setting priorities, a chained connection is used, rather than a bus controller. The enable-in line of the highest-priority processor (SC/MP 1) is tied to the common bus-request line. The enable-out line from SC/MP 1 is tied to the enable-in line of the second SC/MP, and so on. This arrangement provides the chained priority (see also the timing diagram, Fig. 4).



4. Bus access. The SC/MPs are connected in a priority arrangement for access to the common bus. SC/MP 1 has the highest priority, and even if SC/MP 3 has a concurrent request with SC/MP 2, it must wait until SC/MP 2 completes its operations.

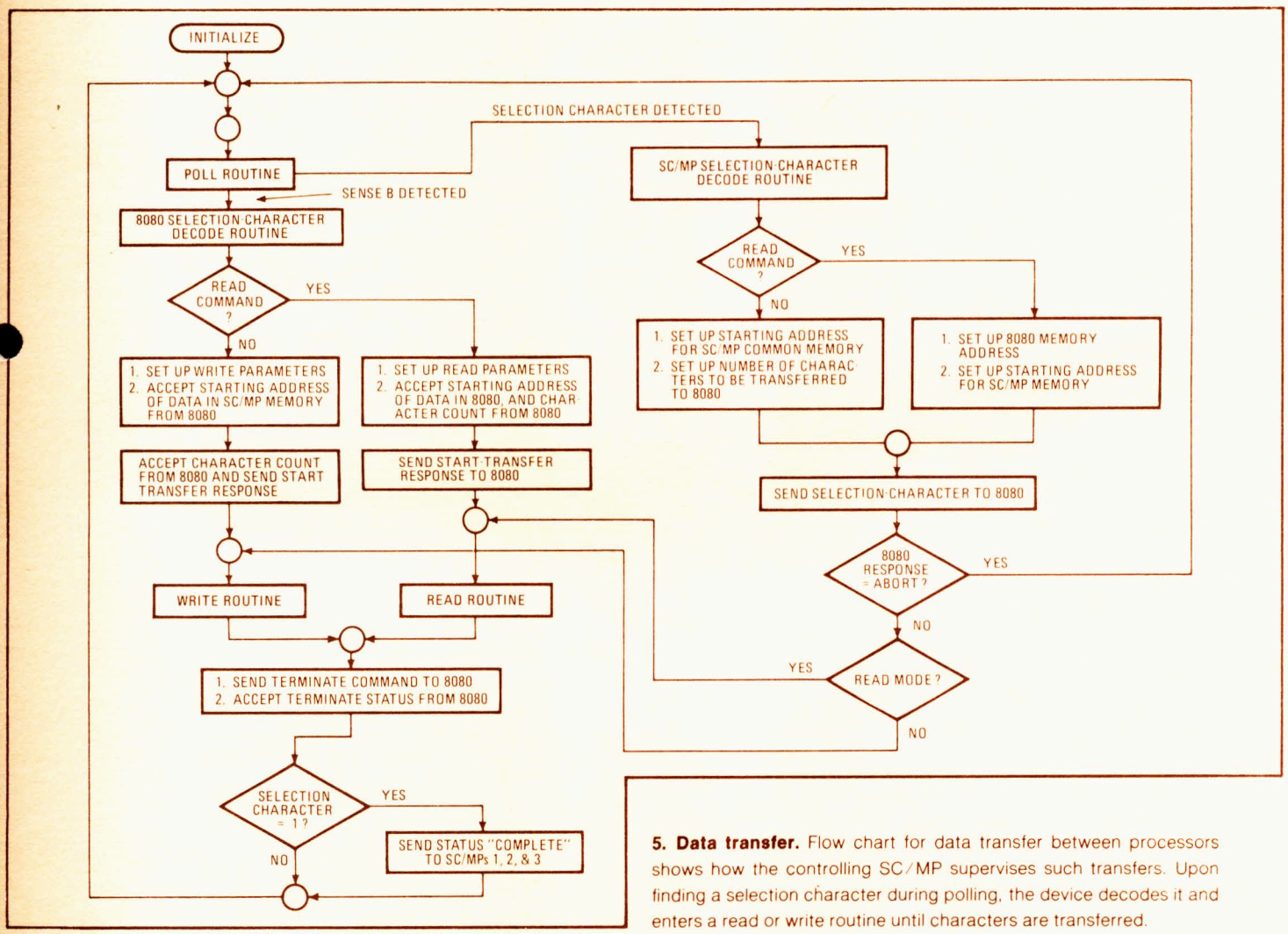
The ENOUT signal from each processor depends on the state of the enable-in line and the internal bus-request latch. When any SC/MP is not using the bus, its enable-out line is held in the same state as its enable-in line.

Since the enable-in line of SC/MP 1 is tied to the common bus-request line, the ENIN signal is activated simultaneously with BREQ. Any other SC/MP that raises a bus request concurrently will not be granted the bus until SC/MP 1 activates the ENOUT signal.

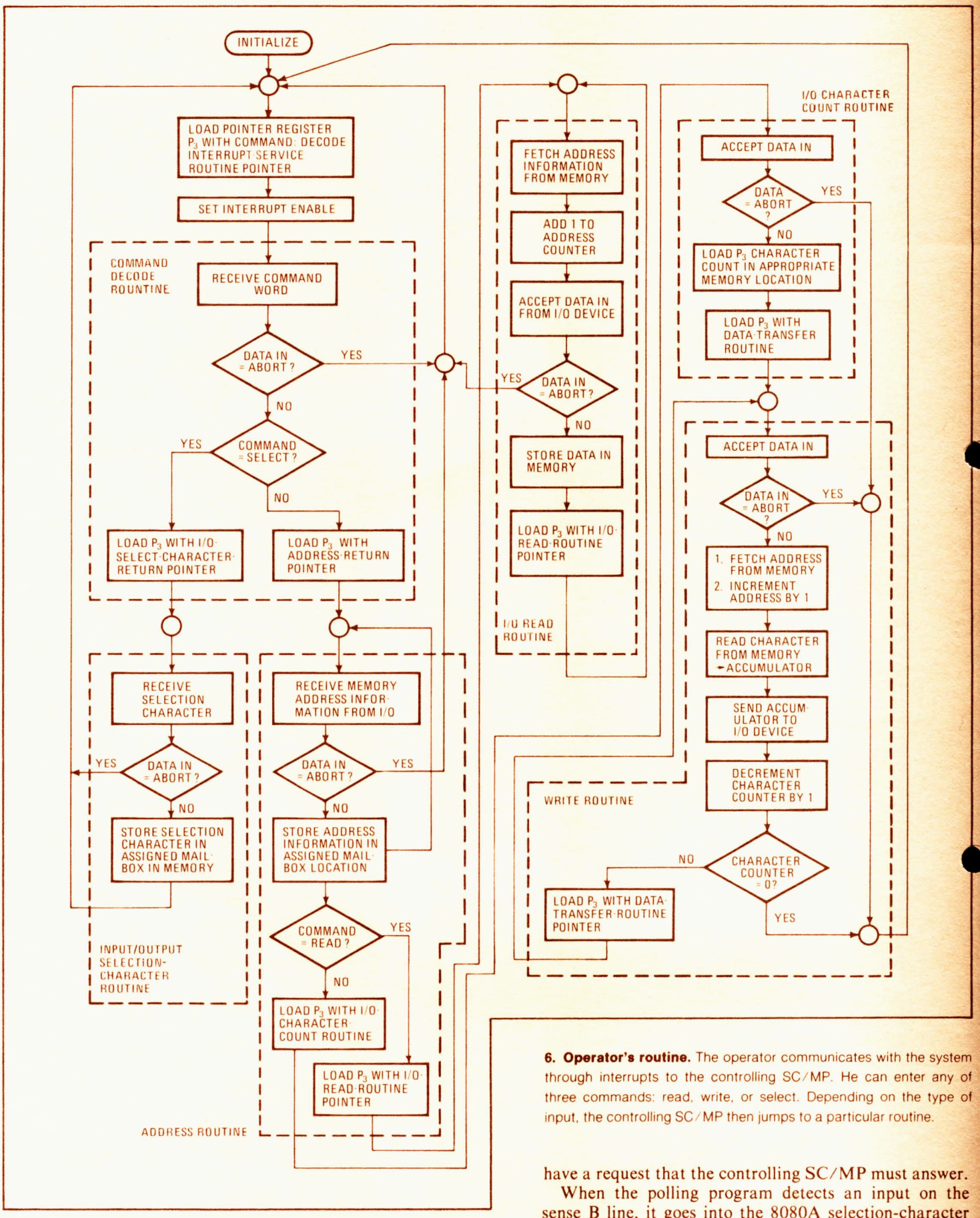
When SC/MP 2 raises the bus request, the ENIN signal to SC/MP 1 is activated. If the internal bus-request line of SC/MP 1 is not set, the ENOUT signal from SC/MP 1 is activated, which activates the ENIN signal of the SC/MP 2. SC/MP 3 must go through both SC/MP 2 and SC/MP 1.

Simple data transfer

The data-transfer program (Fig. 5) in the controlling SC/MP is written so that the microprocessors do not need a complicated routine to request service. (Pointer 3 and the extension register of SC/MP 1 are not available because they are reserved for the interrupt routine associated with the operator's primary peripheral.) The initialize segment of the program sets the address pointers, interrupt enable, and interrupt-service-routine pointer. The initialize routine, first of all, directs each SC/MP in the system to its appropriate program sections in common memory. It also resets some software flags and counters to zero.


INITIALIZE
SELECTION CHARACTER DETECTED
POLL ROUTINE
SC/MP SELECTION-CHARACTER DECODE ROUTINE
SENSE B DETECTED
8080 SELECTION-CHARACTER DECODE ROUTINE
READ COMMAND ?
YES
NO
1. SET UP WRITE PARAMETERS
2. ACCEPT STARTING ADDRESS OF DATA IN SC/MP MEMORY FROM 8080
1. SET UP READ PARAMETERS
2. ACCEPT STARTING ADDRESS OF DATA IN 8080, AND CHARACTER COUNT FROM 8080
ACCEPT CHARACTER COUNT FROM 8080 AND SEND START TRANSFER RESPONSE
SEND START-TRANSFER RESPONSE TO 8080
WRITE ROUTINE
READ ROUTINE
1. SEND TERMINATE COMMAND TO 8080
2. ACCEPT TERMINATE STATUS FROM 8080
SELECTION CHARACTER = 1 ?
SEND STATUS "COMPLETE" TO SC/MPs 1, 2, & 3
1. SET UP STARTING ADDRESS FOR SC/MP COMMON MEMORY
2. SET UP NUMBER OF CHARACTERS TO BE TRANSFERRED TO 8080
1. SET UP 8080 MEMORY ADDRESS
2. SET UP STARTING ADDRESS FOR SC/MP MEMORY
SEND SELECTION-CHARACTER TO 8080
8080 RESPONSE = ABORT ?
READ MODE ?

5. Data transfer. Flow chart for data transfer between processors shows how the controlling SC/MP supervises such transfers. Upon finding a selection character during polling, the device decodes it and enters a read or write routine until characters are transferred.



6. Operator's routine. The operator communicates with the system through interrupts to the controlling SC/MP. He can enter any of three commands: read, write, or select. Depending on the type of input, the controlling SC/MP then jumps to a particular routine.

The controlling SC/MP then enters the polling routine. The 8080A is polled more often than the SC/MPs, since it operates faster and is more likely to have a request that the controlling SC/MP must answer.

When the polling program detects an input on the sense B line, it goes into the 8080A selection-character decode routine. It also scans the mailboxes of each SC/MP to see if selection characters are present. When it detects a selection character in a mailbox, it exits to the proper SC/MP selection-character decode routine.

Decoding selection characters

In the 8080A selection-character decode routine, the controlling SC/MP will read the selection character from the 8212 into its accumulator. The SC/MP then establishes the read or write mode, which is defined relative to the 8080A. It also establishes some address pointers and number of characters to be transferred, based on the mode selected.

The device then sends a start-transfer response to the 8080A, indicating the successful beginning of the data transfer, and exits to the write or read routine. The 8080A, upon sending a selection character to the controlling SC/MP, enters the interrupt mode to communicate all subsequent information. Therefore, its program should have a time-out function so that it will wait indefinitely for a response to its selection character.

The SC/MP selection-character decode routine, like that of the 8080A, establishes the modes, address pointers, number of characters to be transferred, and the starting address of data buffers in SC/MP memory. The controlling SC/MP sends a selection character to the 80A. If it cannot transfer information at this time, the 80A will respond with an abort signal. If it can nsfer information, it responds with a start signal. From this point, the program exits to the read or write routine (here defined relative to the SC/MPs).

In the write routine, data is written from the common memory to the 8080A memory. The controlling SC/MP first loads the starting address of the data to be sent in a pointer register and loads the first data word into the accumulator. Next it sends the first data word to the associated 8212, which in turn sends an interrupt to the 8080A. The SC/MP waits for a response from the 8080A on the sense B input line, decrements the character counter by 1, and increments the address in the pointer register by 1—repeating this process until the content of the character counter is 0.

In the read mode, the routine sets up in the 8080A the starting address of the buffer from which data is to be read. In the SC/MP, the pointer register is loaded with the starting address of the receiving data buffer. The data is read from the 8080A memory into the SC/MP emory as the controlling SC/MP follows a similar utine to that of writing.

The data-transfer termination sequence, which follows the read and write routines, is used to exchange status and other information. If the read or write routine was a SC/MP routine, the termination status should be passed on to the SC/MP that requested the service. Finally, the program loops back to the poll routine to acknowledge the next service requested. Allowing for typical overheads, this sequence will give a transfer rate of about 5,000 characters per second if the new n-channel SC/MPs are used.

The 8080A program is fairly simple, since the burden of controlling data transfers is placed on the controlling SC/MP. However, the program is responsible for creating a selection character whenever the 8080A wants to transfer information. It also is responsible for accepting the interrupts from the controlling SC/MP.

The operator's primary peripheral communicates with the controlling SC/MP on an interrupt basis. The interrupt-handling program (Fig. 6), allows the operator to enter any of three commands: read, write, or select (defined relative to the SC/MP).

Three operator commands

The read command allows data to be entered into the common memory. To avoid loss of data, the operator should be aware of the memory mapping in Fig. 2 while entering the data.

The write command allows the operator to read blocks of data from the common memory. As with the read command, the operator should enter the starting address of the data block he intends to read. In addition, he should also enter the number of characters he wishes to be read from memory.

The select command allows the operator to place a selection character in the mailbox of the controlling SC/MP. As explained, this character will enable the controlling SC/MP to initiate the data-transfer program.

More SC/MPs can be added to the system simply by extending the polling routine and expanding the memory to account for additional mailboxes. Since the SC/MPs have access to the faster peripherals through SC/MP I, they can be made to operate in a virtual-memory mode in which they have a much larger memory available than is directly connected to them. The system also will be useful as a front-end processor for such tasks as editing and setting up of data in a communications systems.

SC/MP LCDS USERS

We have made a change in the monitor program which allows the system to read one character at a time from the TTY paper tape reader. This feature will be required on future software packages.

All systems SN.8017 and below require the new firmware. If you are using the TTY paper tape reader and desire the new ROM. Please send a return address label and serial # to:

National Semiconductor
2900 Semiconductor Drive
Santa Clara, CA. 95051
ATTN: Microcomputer Service MS/206.

P.S. This would also be a good time for you to return the yellow customer report form which came with your LCDS.

SC/MP-II IS HERE!

The revolutionizing SC/MP-II is faster, operates only with single +5V supply with power dissipation ≤200MW and requires very low cost, 3.58 MHZ or 4.00 MHZ timing crystal.

SC/MP-II is software and pinout compatible with SC/MP. SC/MP-II is available in low cost molded dip, i.e. plastic as well as ceramic package.

Suggested prices as follows:

	1 up	25 up	100 up
ISP-8A/600N (Plastic Pkg.)	$ 14.92	$ 12.00	$ 9.00
ISP-8A/600D (Ceramic Pkg.)	$ 17.76	$ 16.00	$ 15.00

For information on how to upgrade your SC/MP Kit, see page 19.

1976 BIT BUCKET/COMPUTE INDEX

Articles

A Data Concentrator using PACE;
by Barney Hordos, July, page 14

Additions to the Subroutine Library;
December, page 7

A High Level Language for IMP-16;
August, page 1

An IMP-16 Microprocessor System Part 1;
by Hal Chamberlin, June, page 1

An IMP-16 Microprocessor System Part 2;
by Hal Chamberlin, July, page 17

An IMP-16 Microprocessor System Part 3;
by Hal Chamberlin, November, page 1

A Poet's Corner. . ?;
by Ira McComic, February, page 7

Application Corner, High Performance,
Low Power Memories from Inexpensive Parts;
November, page 18

A Quick and Dirty Transfer Method;
by William E. Homes, March, page 6

A 16 Bit Computer Kit Evaluation;
by John Snell, August, page 1

Back by Popular Demand: SC/MP;
by Dave Graves, July, page 2

Build Your Own BASIC;
by Dennis Allison and others, April, page 2

Cards and Controllers;
April, page 6

Commands & Controls;
February, page 8

Communication Interfaces for the IMP-16;
by Al Weisberger, January, page 1

COMPUTE Coding Problem Number 1;
February, page 1

Computer Publications;
January, page 6

Contest Winner;
June, page 5

Cross Assembler for IMP-16, PACE;
April, page 6

Disc Software Update;
January, page 5

Edit Commands;
January, page 5

Errata;
February, page 3; May, page 8;
June, page 15; August, page 9

Good Reading;
April, page 1

Hamilton/Avnet uses PACE in New Kit;
January, page 1

IMP-16P and PACE Teletype I/O Routines;
February, page 7

IMP-16/CRT DOS;
July, page 7

Increasing Throughput for IMP-16 Serial Input/Output;
by Keith Winter, September, page 1

Interfacing a Keyboard to SC/MP;
December, page 2

Loading CASM on the Disc;
October, page 6

Low-Cost Development System for SC/MP Microprocessor;
October, page 5

Memory Addressing in PACE Microprocessor;
August, page 6

Memory at a Glance;
October, page 4

Microprocessor Consultant and Product Announcement;
July, page 4

Microprocessor Consultants;
April, page 11; September, page 7

Microprocessors from National;
March, page 14

Microprocessor Simplify Industrial Control Systems;
by Alan J. Weissberger, page 3

MuPro 8080 Development System is Now Available;
December, page 1

National Announces "8080A" Microprocessor;
July, page 1

National's New Low-Cost Microprocessor;
March, page 1

Need Application Help;
April, page 1

New IMP-16 Interface Cards;
April, page 1

New PACE PC Board Vendor;
April, page 11

National Semiconductor

DECEMBER 1976
Pub. No. 426305326-001A

INS8255 Programmable Peripheral Interface

general description

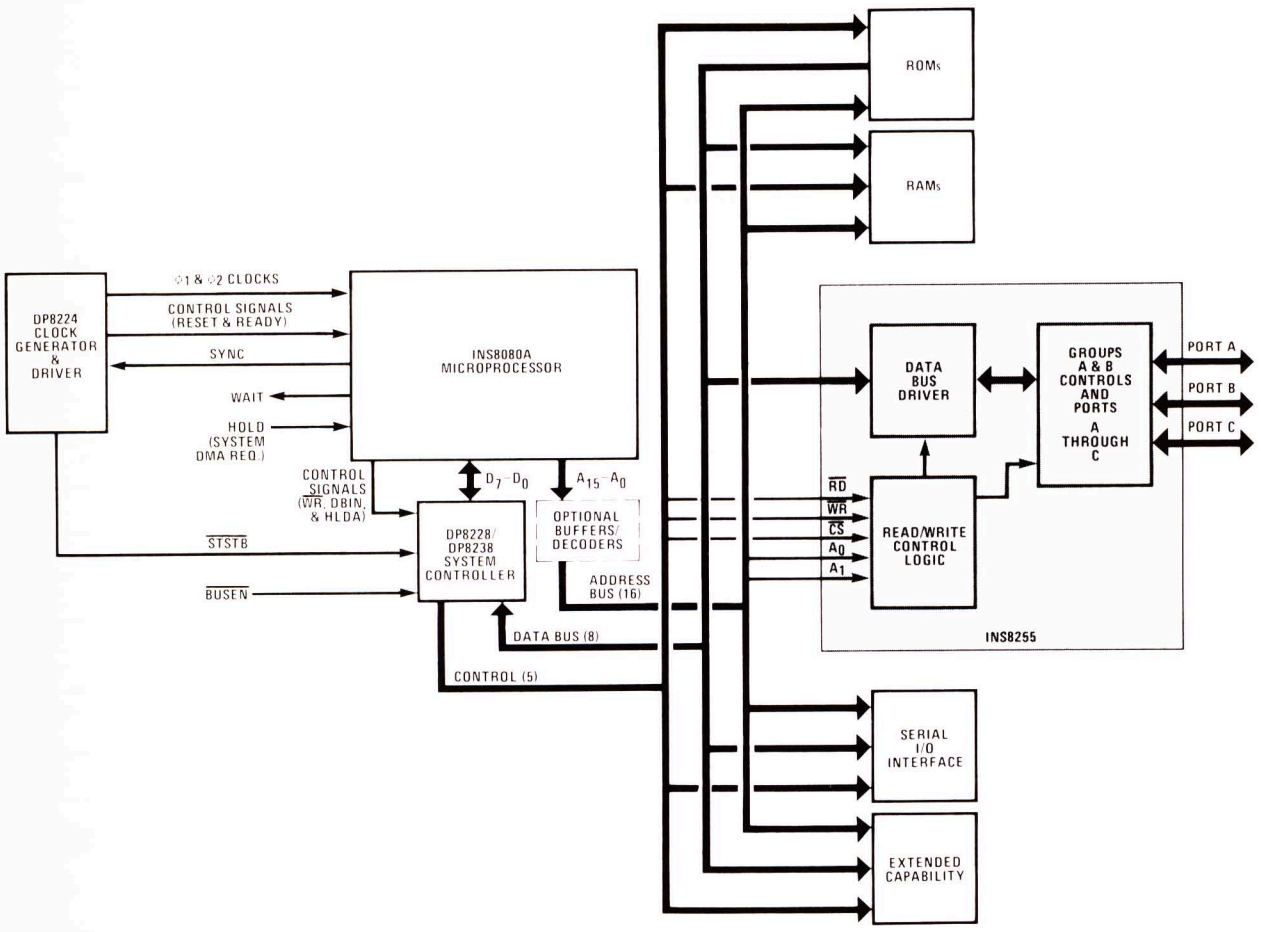
The INS8255 is a programmable peripheral interface contained in a standard, 40-pin dual-in-line package. The chip, which is fabricated using N-channel silicon gate technology, functions as a general-purpose parallel input/output interface in National Semiconductor's N8080 microcomputer family. The functional configuration of the INS8255 is programmed by the system software so that normally no external logic is required to interface peripheral devices.

The INS8255 has three basic modes of operation that can be selected by the system software. In the first mode (Mode 0), the INS8255 provides simple input and output operations for three 8-bit ports. Data is simply written to or read from a specified port (Port A, B or C) without the use of "handshaking" signals. In the second mode (Mode 1), the INS8255 enables the transfer of input/output data to or from a specified 8-bit port (Port A or B) in conjunction with strobes or "handshaking" signals. Ports A and B use the lines of Port C in this mode to generate or accept the "handshaking" signals with the peripheral device. In the third mode (Mode 2), the INS8255 enables communications with a peripheral device or structure via one bidirectional 8-bit bus port (Port A). "Handshaking" signals are provided over the lines of Port C in this mode to maintain proper bus flow discipline.

features

- Outputs Source 1 mA at 1.5 Volts
- 24 Programmable Input/Output Pins
- Direct Bit Set/Reset Capability
- TTL Compatible
- Reduces System Component Count

N8080A microcomputer family block diagram



© 1976 National Semiconductor Corp.

DA-B20M126/Printed in U.S.A.

dc electrical characteristics

$T_A = 0°C$ to $+70°C$; $V_{CC} = +5\,V \pm 5\%$; $V_{SS} = 0\,V$

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage			0.8	V	
V_{IH}	Input High Voltage	2.0			V	
V_{OL}	Output Low Voltage			0.4	V	$I_{OL} = 1.6\,mA$
V_{OH}	Output High Voltage	2.4			V	$I_{OH} = -50\,\mu A$ ($-100\,\mu A$ for D.B. Port)
I_{OH}[1]	Darlington Drive Current		2.0		mA	$V_{OH} = 1.5\,V$, $R_{EXT} = 390\,\Omega$
I_{CC}	Power Supply Current		40		mA	

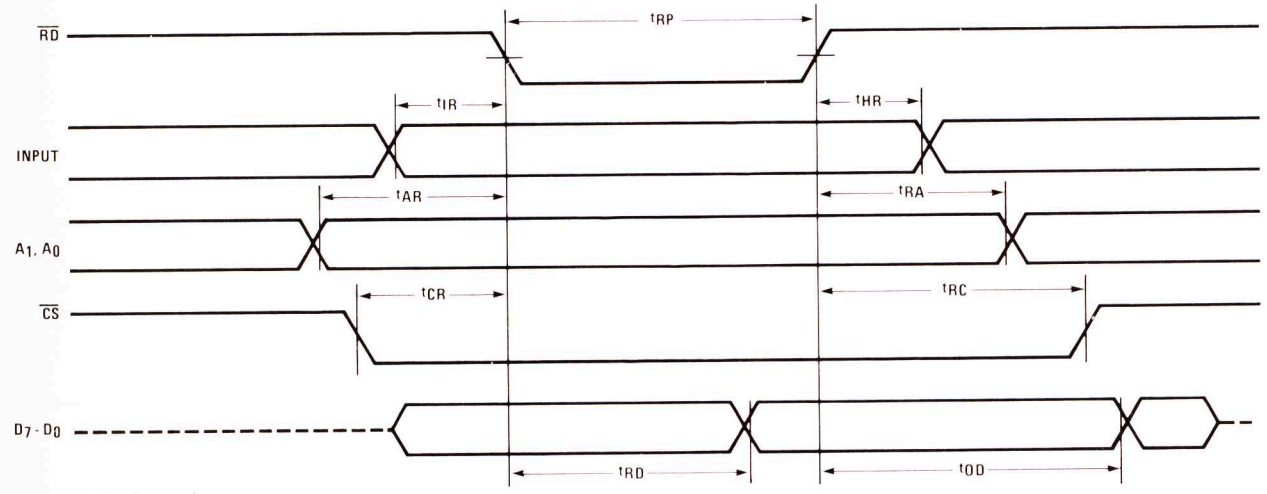
NOTE:
1. Available on 8 pins only of ports A and C. Selected randomly.

ac electrical characteristics

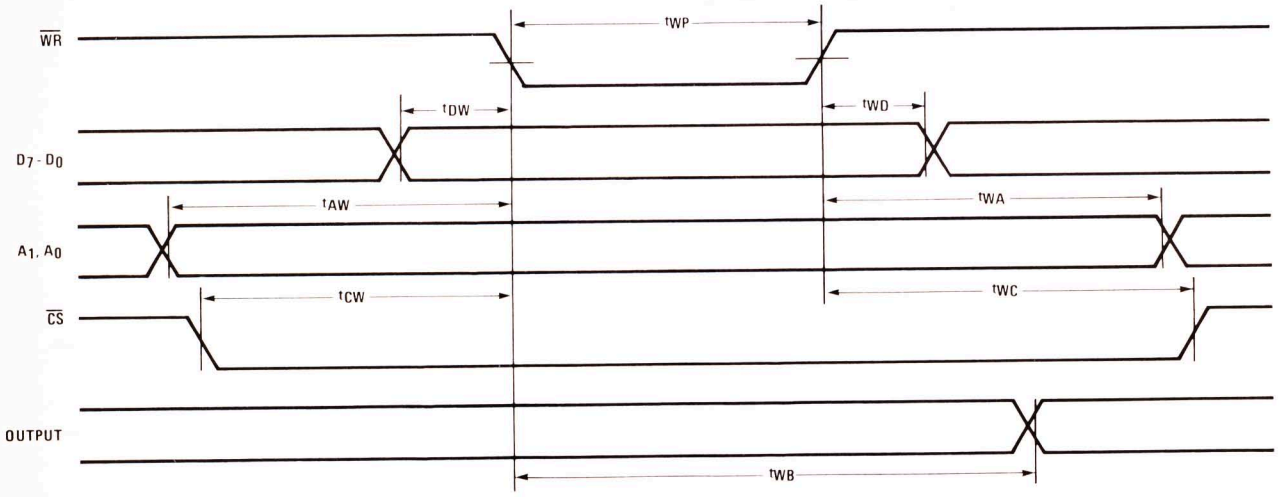
$T_A = 0°C$ to $+70°C$; $V_{CC} = +5\,V \pm 5\%$; $V_{SS} = 0\,V$

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
t_{WP}	Pulse Width of $\overline{WR}$	430			ns	
t_{DW}	Time D.B. Stable before $\overline{WR}$	10			ns	
t_{WD}	Time D.B. Stable after $\overline{WR}$	65			ns	
t_{AW}	Time Address Stable before $\overline{WR}$	20			ns	
t_{WA}	Time Address Stable after $\overline{WR}$	35			ns	
t_{CW}	Time CS Stable before $\overline{WR}$	20			ns	
t_{WC}	Time CS Stable after $\overline{WR}$	35			ns	
t_{WB}	Delay from $\overline{WR}$ to Output			500	ns	
t_{RP}	Pulse Width of $\overline{RD}$	430			ns	
t_{IR}	$\overline{RD}$ Set-Up Time	50			ns	
t_{HR}	Input Hold Time	50			ns	
t_{RD}	Delay from $\overline{RD} = 0$ to System Bus			350	ns	
t_{OD}	Delay from $\overline{RD} = 1$ to System Bus	150			ns	
t_{AR}	Time Address Stable before $\overline{RD}$	50			ns	
t_{CR}	Time $\overline{CS}$ Stable before $\overline{RD}$	50			ns	
t_{AK}	Width of $\overline{ACK}$ Pulse	500			ns	
t_{ST}	Width of $\overline{STB}$ Pulse	350			ns	
t_{PS}	Set-Up Time for Peripheral	150			ns	
t_{PH}	Hold Time for Peripheral	150			ns	
t_{RA}	Hold Time for A_1, A_0 after $\overline{RD} = 1$	379			ns	
t_{RC}	Hold Time for $\overline{CS}$ after $\overline{RD} = 1$	5			ns	
t_{AD}	Time from $\overline{ACK} = 0$ to Output (Mode 2)			500	ns	
t_{KD}	Time from $\overline{ACK} = 1$ to Output Floating			300	ns	
t_{WO}	Time from $\overline{WR} = 1$ to $\overline{OBF} = 0$			300	ns	
t_{AO}	Time from $\overline{ACK} = 0$ to $\overline{OBF} = 1$			500	ns	
t_{SI}	Time from $\overline{STB} = 0$ to IBF			600	ns	
t_{RI}	Time from $\overline{RD} = 1$ to IBF = 0			300	ns	

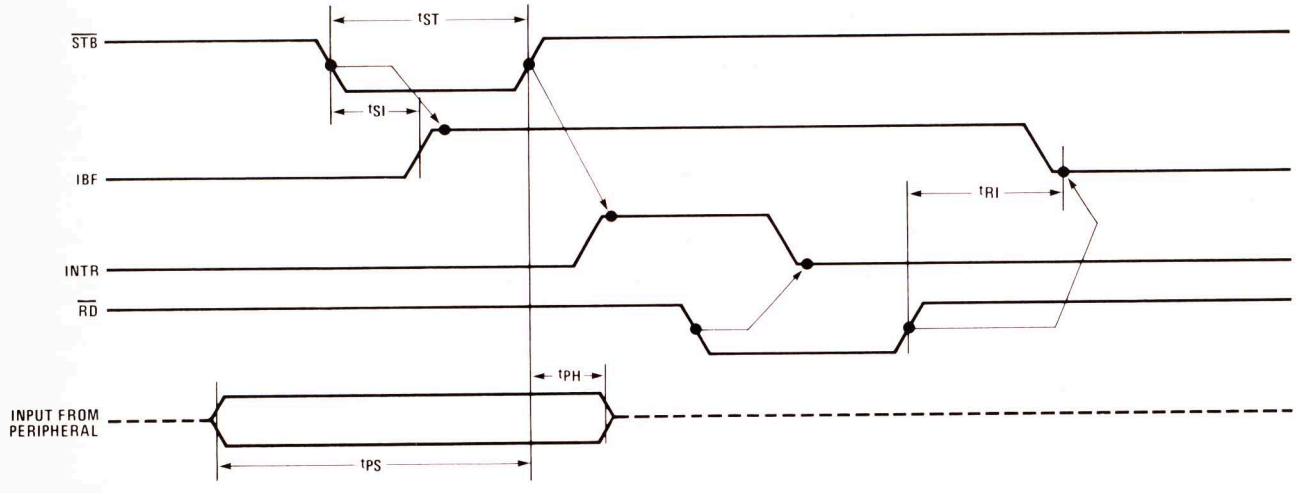
timing waveforms



mode 0 (basic input)

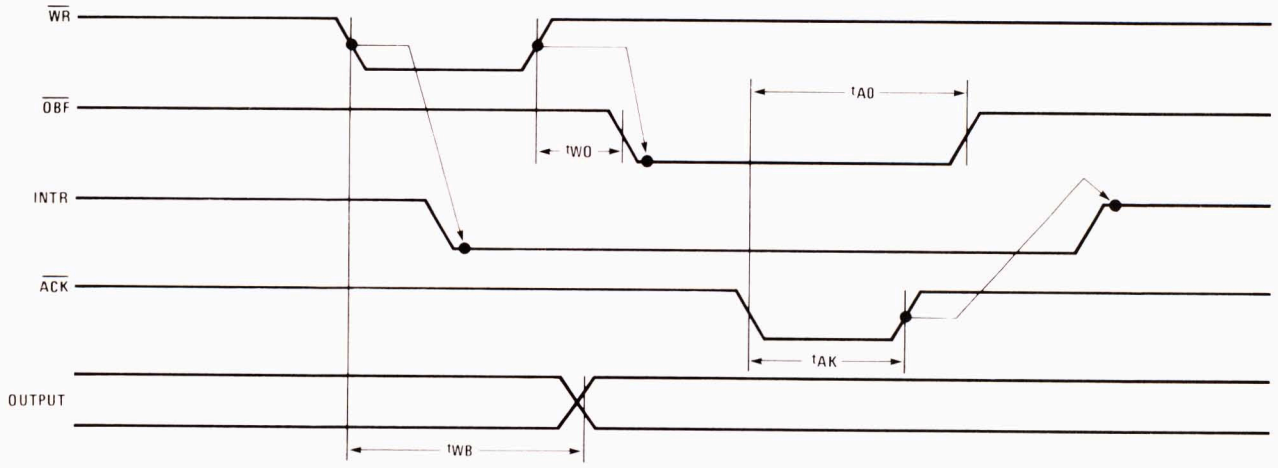


mode 0 (basic output)

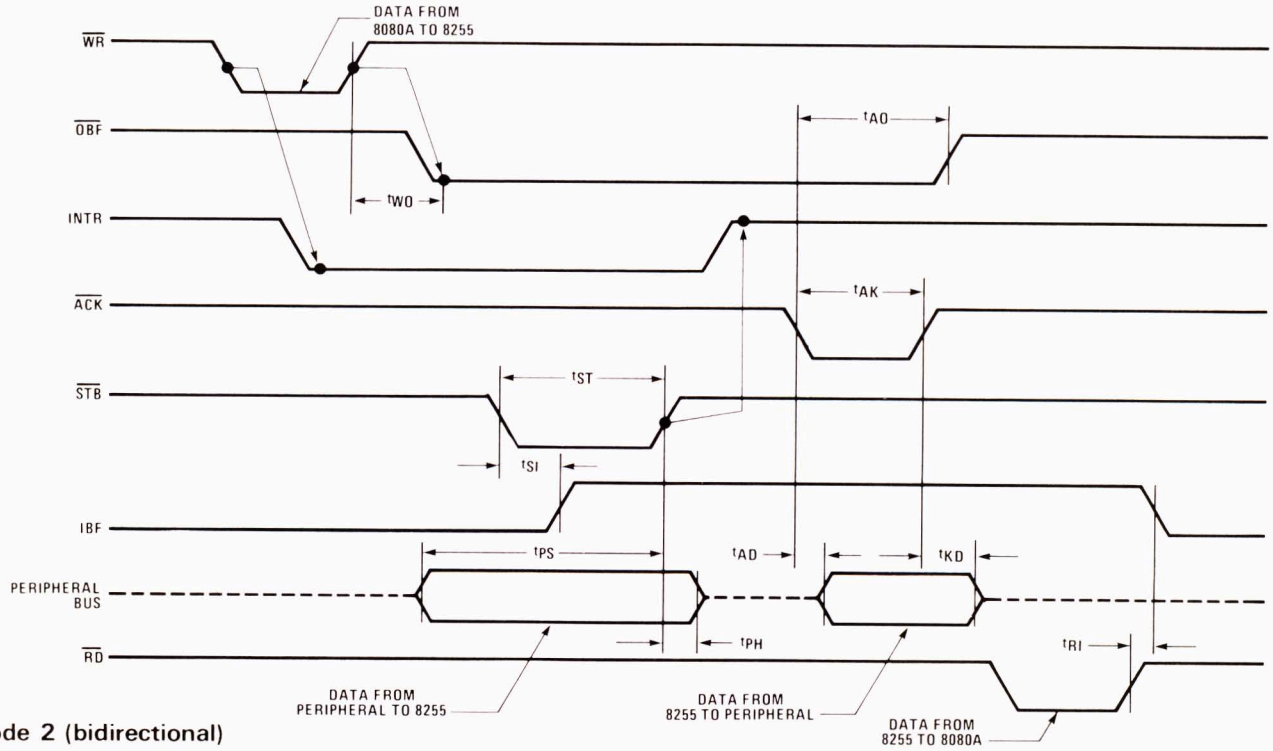


mode 1 (strobed input)

timing waveforms (cont'd.)

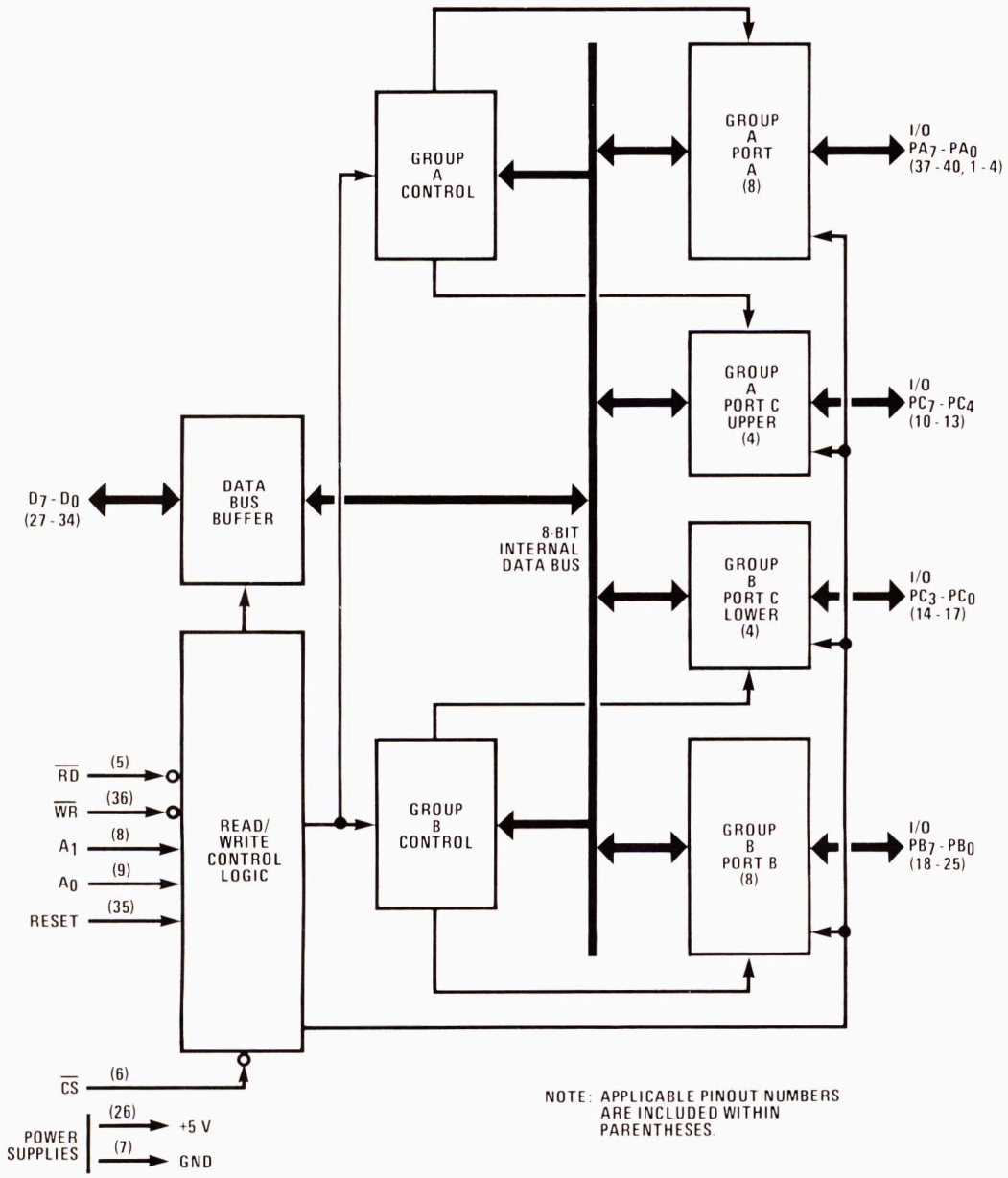


mode 1 (strobed output)



mode 2 (bidirectional)

INS8255 block diagram


GROUP A CONTROL
GROUP A PORT A (8)
I/O PA7 - PA0 (37 - 40, 1 - 4)
GROUP A PORT C UPPER (4)
I/O PC7 - PC4 (10 - 13)
D7 - D0 (27 - 34)
DATA BUS BUFFER
8-BIT INTERNAL DATA BUS
GROUP B PORT C LOWER (4)
I/O PC3 - PC0 (14 - 17)
RD (5)
WR (36)
A1 (8)
A0 (9)
RESET (35)
READ/ WRITE CONTROL LOGIC
GROUP B CONTROL
GROUP B PORT B (8)
I/O PB7 - PB0 (18 - 25)
CS (6)
POWER SUPPLIES
(26) +5 V
(7) GND
NOTE: APPLICABLE PINOUT NUMBERS ARE INCLUDED WITHIN PARENTHESES.

INS8255 functional pin definitions

The following describes the function of all the INS8255 input/output pins. Some of these descriptions reference internal circuits.

INPUT SIGNALS

Chip Select ($\overline{CS}$): When low, the chip is selected. This enables communication between the INS8255 and the INS8080A microprocessor.

Read ($\overline{RD}$): When low, allows the INS8080A to read data or status information from the INS8255.

Write ($\overline{WR}$): When low, allows the INS8080A to write data or control words into the INS8255.

Port Select (A_0, A_1): These two inputs, which are normally connected to the least significant bits of the A_{15} - A_0 Address Bus, control the selection of one of three 8-bit ports (A, B and C) or the internal control word register as indicated below.

A_1	A_0	Selected
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Control Word Register

Reset: When high, clears all the internal registers of the chip and sets Ports A, B and C to the input high impedance mode.

+5 Volts: V_{CC} Supply.

Ground: 0-Volt Reference.

INPUT/OUTPUT SIGNALS

Data (D_7 - D_0) Bus: This bus comprises eight TRI-STATE input/output lines. The bus provides bidirectional communication between the INS8255 and the INS8080A. Data is routed to or from the internal data bus buffer upon execution of an OUT or IN Instruction, respectively, by the INS8080A. In addition, control words and status information are transferred through the data bus buffer.

Port A (PA_7 - PA_0): This 8-bit input/output port comprises one 8-bit data output latch/buffer and one 8-bit data input latch.

NOTE

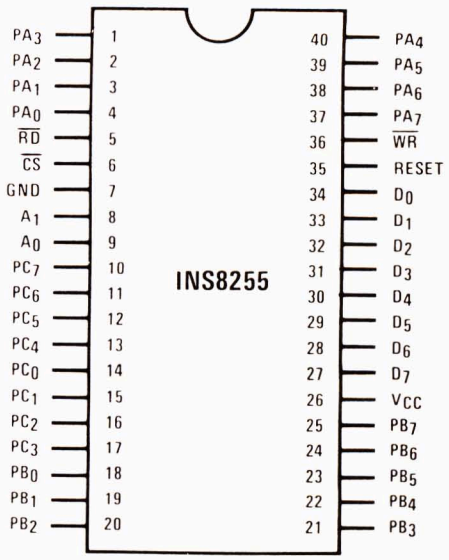
The system software uses a Mode Definition Control Word (see figure) as the second byte of OUT Instruction(s) to program the functional configuration of Ports A through C. Whenever the mode is changed, all output registers (and status flip-flops) are reset.

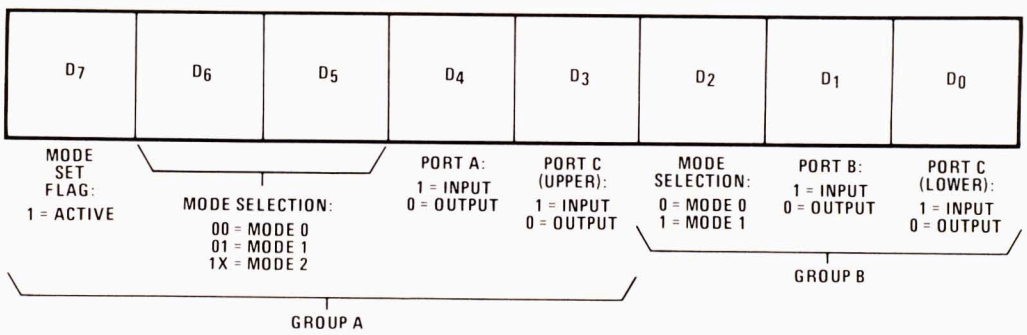
Port B (PB_7 - PB_0): This 8-bit input/output port comprises one 8-bit data input and output latch/buffer and one 8-bit data input buffer.

Port C (PC_7 - PC_0): This 8-bit input/output port comprises one 8-bit data output latch/buffer and one 8-bit data input buffer. The port can be split into two 4-bit ports under the mode control. Each of these 4-bit ports contains a 4-bit latch that may be used for the control and status signals, in conjunction with Ports A and B.

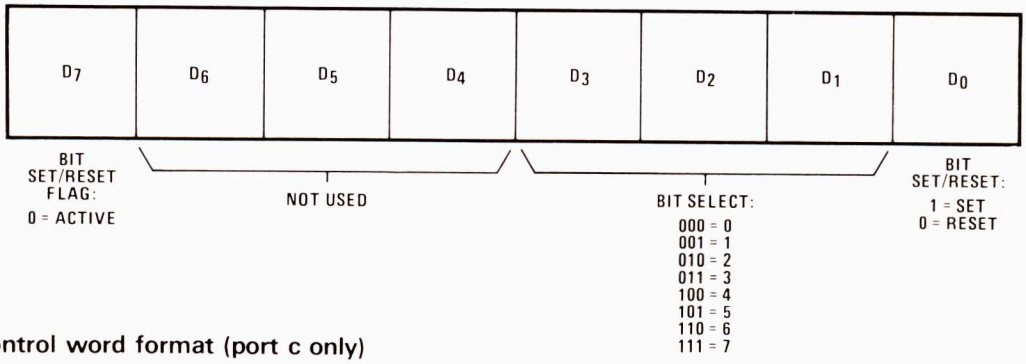
The system software includes a Bit Set/Reset Control Word (see figure) for setting or resetting any of the eight bits of Port C. When Port C is being used as a status/control for Port A or B, the Port C bits can be set or reset by using the Bit Set/Reset Control Word as the second byte of OUT Instruction(s).

pin configuration




D7 D6 D5 D4 D3 D2 D1 D0
MODE SET FLAG: 1 = ACTIVE
MODE SELECTION: 00 = MODE 0 01 = MODE 1 1X = MODE 2
PORT A: 1 = INPUT 0 = OUTPUT
PORT C (UPPER): 1 = INPUT 0 = OUTPUT
MODE SELECTION: 0 = MODE 0 1 = MODE 1
PORT B: 1 = INPUT 0 = OUTPUT
PORT C (LOWER): 1 = INPUT 0 = OUTPUT
GROUP A
GROUP B

mode definition control word format


D7 D6 D5 D4 D3 D2 D1 D0
BIT SET/RESET FLAG: 0 = ACTIVE
NOT USED
BIT SELECT: 000 = 0 001 = 1 010 = 2 011 = 3 100 = 4 101 = 5 110 = 6 111 = 7
BIT SET/RESET: 1 = SET 0 = RESET

bit set/reset control word format (port c only)

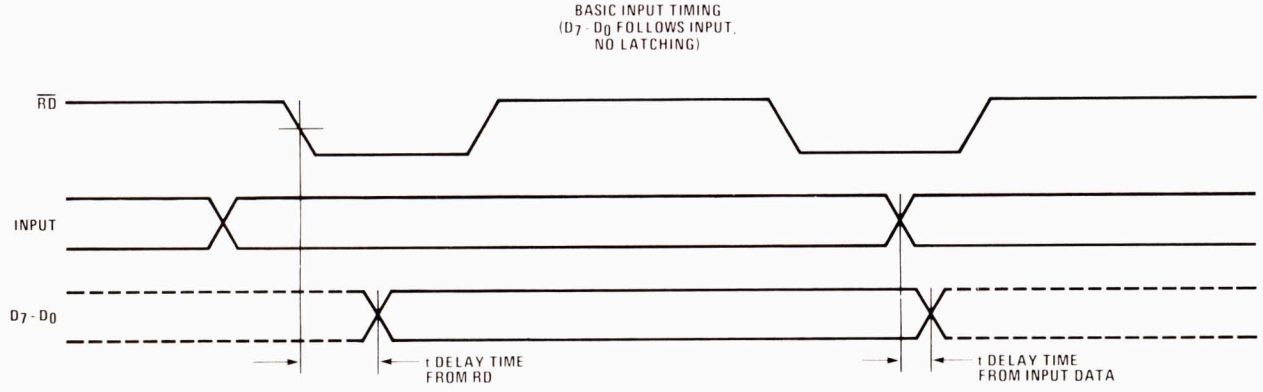
operating modes

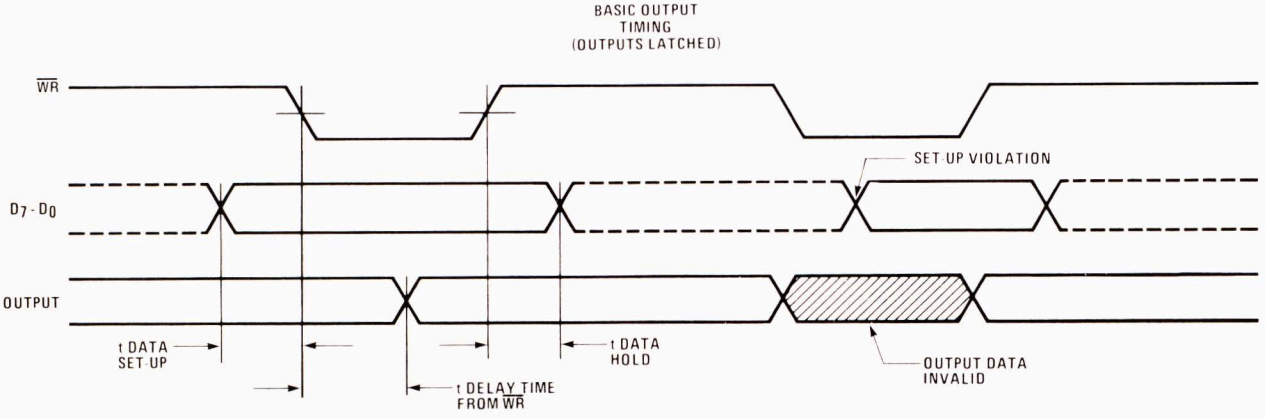
mode 0 (basic input/output)

In this mode, simple input and output operations for each of the three ports are provided. No "handshaking" is required; data is simply written to or read from a specified port.

mode 0 port definition chart

No.	Control Word Bits								Group A		Group B	
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Port A	Port C (Upper)	Port B	Port C (Lower)
0	1	0	0	0	0	0	0	0	OUTPUT	OUTPUT	OUTPUT	OUTPUT
1	1	0	0	0	0	0	0	1	OUTPUT	OUTPUT	OUTPUT	INPUT
2	1	0	0	0	0	0	1	0	OUTPUT	OUTPUT	INPUT	OUTPUT
3	1	0	0	0	0	0	1	1	OUTPUT	OUTPUT	INPUT	INPUT
4	1	0	0	0	1	0	0	0	OUTPUT	INPUT	OUTPUT	OUTPUT
5	1	0	0	0	1	0	0	1	OUTPUT	INPUT	OUTPUT	INPUT
6	1	0	0	0	1	0	1	0	OUTPUT	INPUT	INPUT	OUTPUT
7	1	0	0	0	1	0	1	1	OUTPUT	INPUT	INPUT	INPUT
8	1	0	0	1	0	0	0	0	INPUT	OUTPUT	OUTPUT	OUTPUT
9	1	0	0	1	0	0	0	1	INPUT	OUTPUT	OUTPUT	INPUT
10	1	0	0	1	0	0	1	0	INPUT	OUTPUT	INPUT	OUTPUT
11	1	0	0	1	0	0	1	1	INPUT	OUTPUT	INPUT	INPUT
12	1	0	0	1	1	0	0	0	INPUT	INPUT	OUTPUT	OUTPUT
13	1	0	0	1	1	0	0	1	INPUT	INPUT	OUTPUT	INPUT
14	1	0	0	1	1	0	1	0	INPUT	INPUT	INPUT	OUTPUT
15	1	0	0	1	1	0	1	1	INPUT	INPUT	INPUT	INPUT

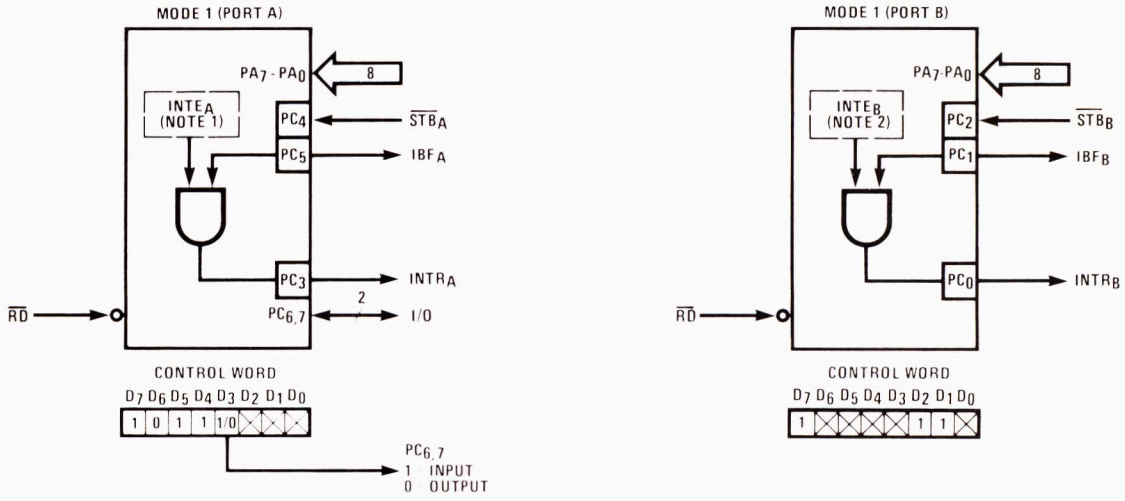




mode 0 timing

mode 1 (strobed input/output)

In this mode, a means for transferring input/output data to or from a specified port in conjunction with strobes or "handshaking" signals is provided. Port A and Port B use the lines on Port C to generate or accept these "handshaking" signals in Mode 1. The programmer can read the contents of Port C to test or verify the status of each peripheral device. Since no special instruction is provided in the INS8080A microcomputer system to read the Port C status information, a normal read operation must be executed to perform this function.

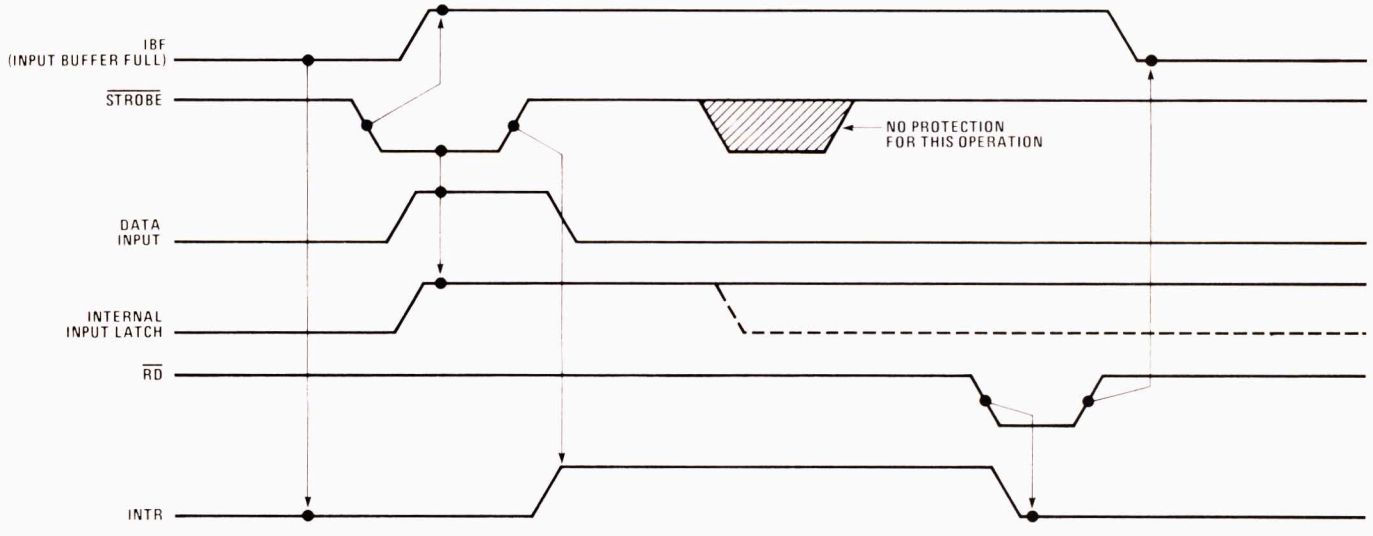


Notes:
1. $INTE_A$ is controlled by bit set/reset of PC_4.
2. $INTE_B$ is controlled by bit set/reset of PC_2.

STATUS WORD

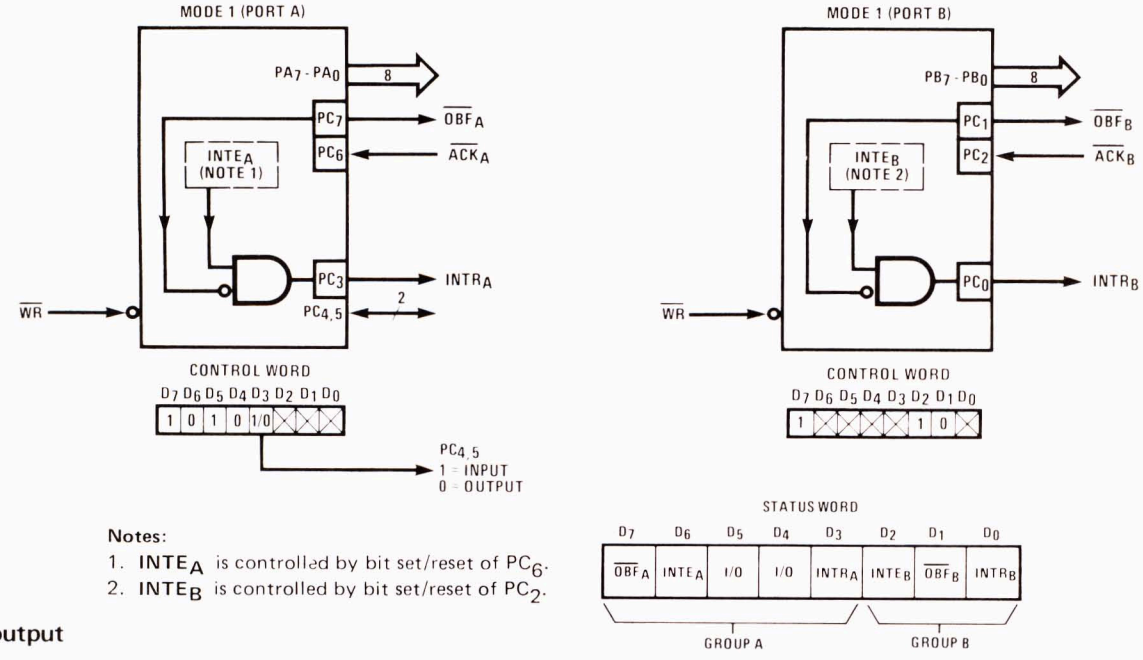
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
GROUP A					GROUP B		

mode 1 input



mode 1 input timing

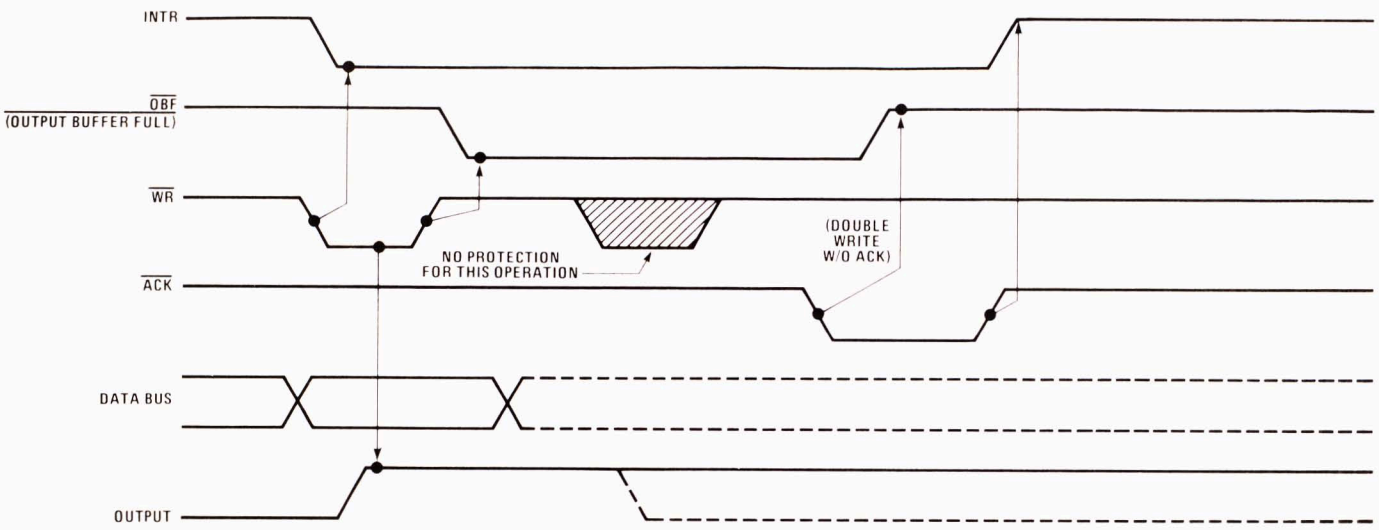
operating modes (cont'd.)



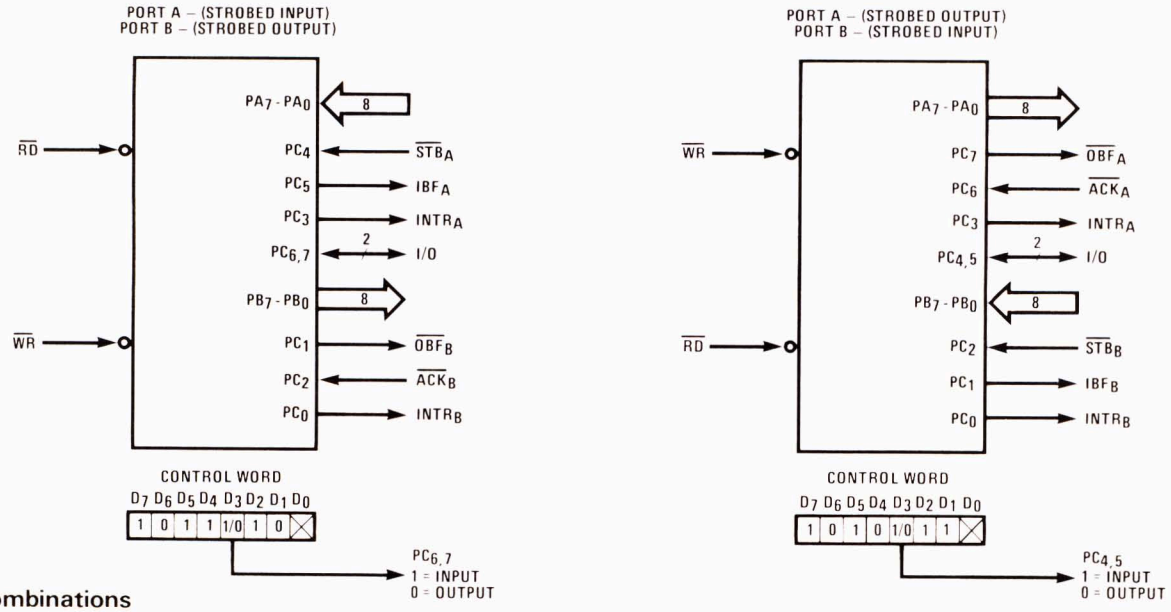
Notes:

1. $INTE_A$ is controlled by bit set/reset of PC_6.
2. $INTE_B$ is controlled by bit set/reset of PC_2.

mode 1 output



mode 1 output timing

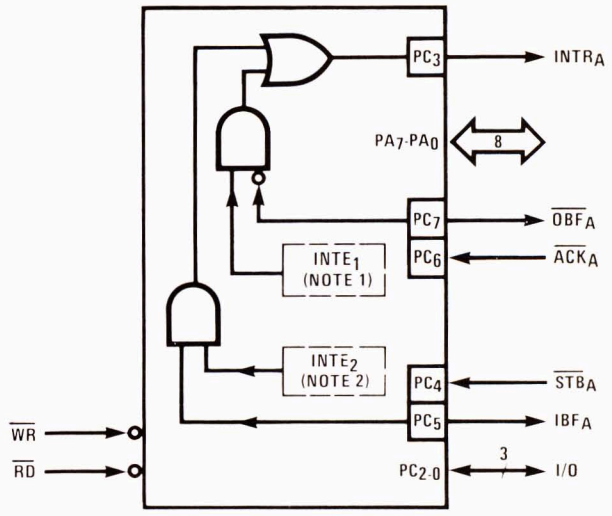


mode 1 combinations

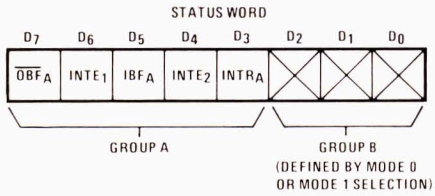
operating modes (cont'd.)

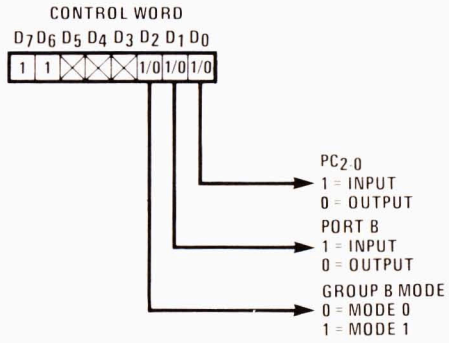
mode 2 (strobed bidirectional bus input/output)

This mode enables communication with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus input/output). "Handshaking" signals are provided to maintain proper bus flow discipline in a manner similar to Mode 1. In addition, interrupt generation and enable/disable functions are available in Mode 2.

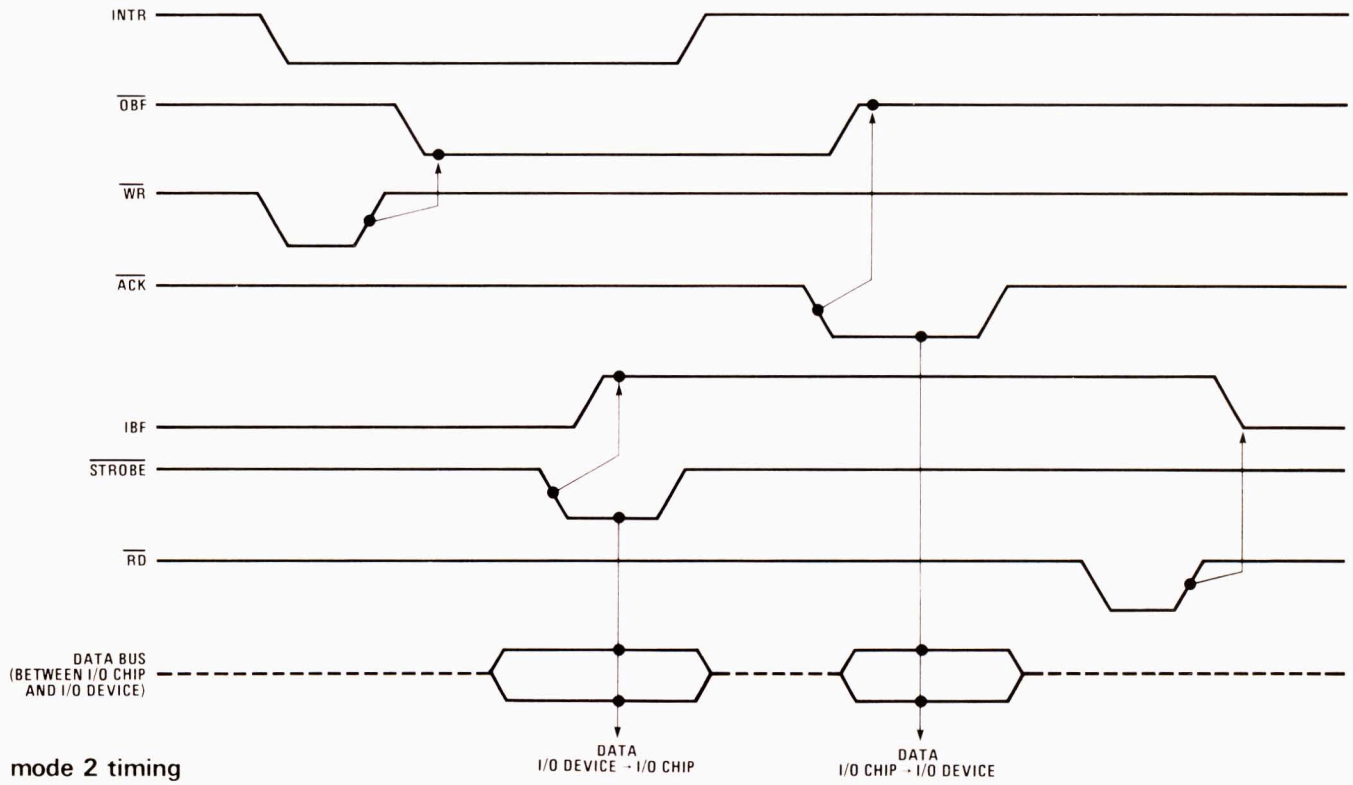


Notes:
1. $INTE_1$ is controlled by bit set/reset of PC_6.
2. $INTE_2$ is controlled by bit set/reset of PC_4.





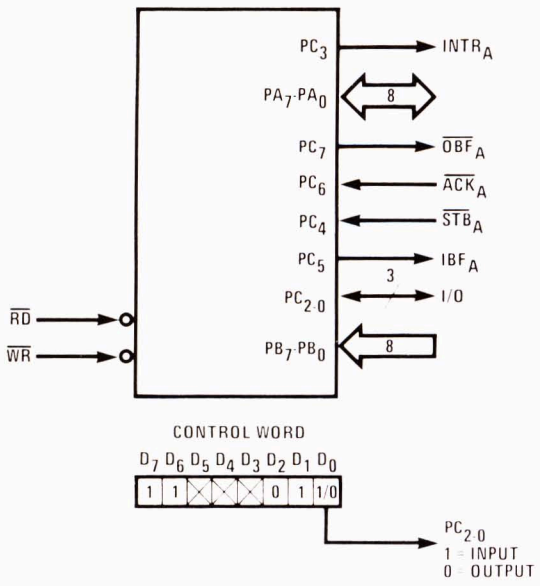
mode 2



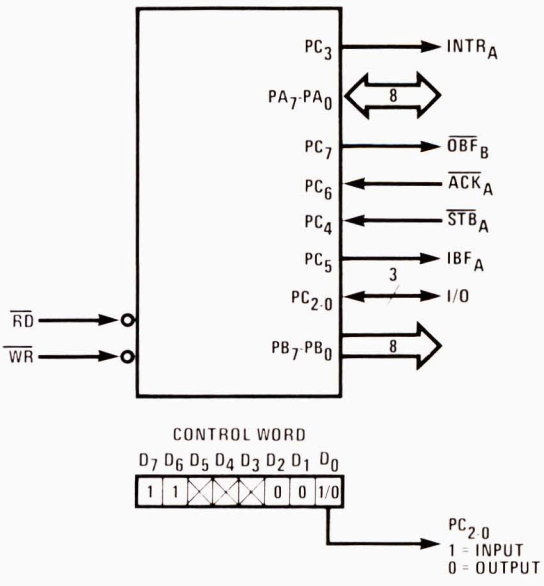
mode 2 timing

operating modes (cont'd.)

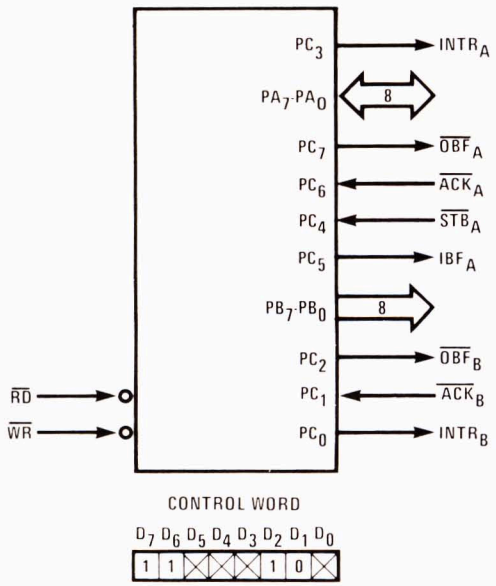
MODE 2 AND MODE 0 (INPUT)



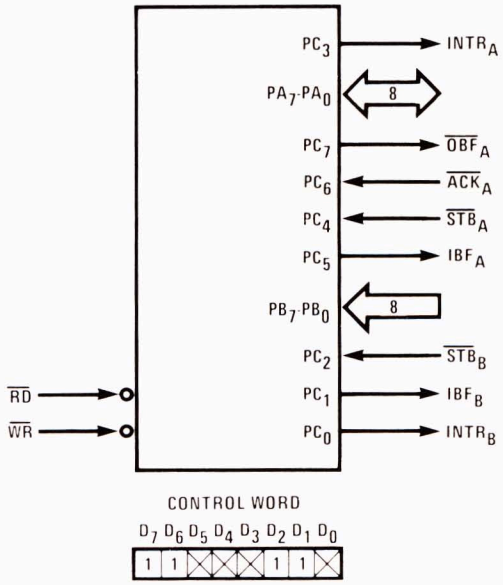
MODE 2 AND MODE 0 (OUTPUT)



MODE 2 AND MODE 1 (OUTPUT)



MODE 2 AND MODE 1 (INPUT)

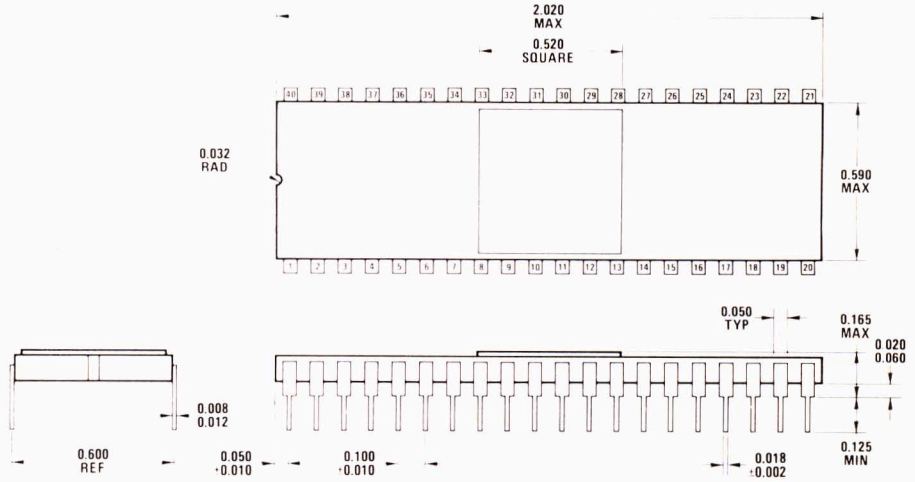


mode 2 combinations

mode definition summary table

Port Bits	Mode 0		Mode 1		Mode 2
	IN	OUT	IN	OUT	Group A Only
PA_0	IN	OUT	IN	OUT	Bidirectional
PA_1	IN	OUT	IN	OUT	↕
PA_2	IN	OUT	IN	OUT	↕
PA_3	IN	OUT	IN	OUT	↕
PA_4	IN	OUT	IN	OUT	↕
PA_5	IN	OUT	IN	OUT	↕
PA_6	IN	OUT	IN	OUT	↕
PA_7	IN	OUT	IN	OUT	Bidirectional
PB_0	IN	OUT	IN	OUT	
PB_1	IN	OUT	IN	OUT	
PB_2	IN	OUT	IN	OUT	
PB_3	IN	OUT	IN	OUT	
PB_4	IN	OUT	IN	OUT	(Mode 0 or Mode 1 only)
PB_5	IN	OUT	IN	OUT	
PB_6	IN	OUT	IN	OUT	
PB_7	IN	OUT	IN	OUT	
PC_0	IN	OUT	$INTR_B$	$INTR_B$	I/O
PC_1	IN	OUT	IBF_B	$\overline{OBF}_B$	I/O
PC_2	IN	OUT	$\overline{STB}_B$	$\overline{ACK}_B$	I/O
PC_3	IN	OUT	$INTR_A$	$INTR_A$	$INTR_A$
PC_4	IN	OUT	$\overline{STB}_A$	I/O	$\overline{STB}_A$
PC_5	IN	OUT	IBF_A	I/O	IBF_A
PC_6	IN	OUT	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$
PC_7	IN	OUT	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$

physical dimensions



Ceramic Dual-In-Line Package (D)
Order Number INS8255D

Manufactured under one or more of the following U.S. patents: 3083262, 3189758, 3231797, 3303356, 3317671, 3323071, 3381071, 3408542, 3421025, 3426423, 3440498, 3518750, 3519897, 3557431, 3560765, 3566216, 3571630, 3575609, 3579059, 3580969, 3597640, 3607469, 3617859, 3631312, 3633052, 3638131, 3648071, 3651565, 3693248.

National Semiconductor Corporation
2900 Semiconductor Drive, Santa Clara, California 95051, (408) 737-5000/TWX (910) 339-9240
National Semiconductor GmbH
808 Fuerstenfeldbruck, Industriestrasse 10, West Germany, Tele. (08141) 1371/Telex 05-27649
National Semiconductor (UK) Ltd.
Larkfield Industrial Estate, Greenock, Scotland, Tele. (0475) 33251/Telex 778-632

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied, and National reserves the right, at any time without notice, to change said circuitry.

New Publications;
October, page 11

New Product Information;
August, page 6

Other Clubs and Publications;
July, page 11

Other Publications;
February, page 3

Overseas User Library Sources;
January, page 5

PACE Instruction Set;
July, page 9

PACE Problems and Corrections;
June, page 2

PACE Software and Documentation;
February, page 7

PACRAM Update;
April, page 11

People's Computer Company;
January, page 6

Processor is Calculator-Oriented;
September, page 10

Program Library News;
July, page 11

Programming Tidbits;
October, page 4; December, page 4

PROM Erasing Equipment;
April, page 1

Prototyping Board;
July, page 4

Recent Literature;
January, page 4

SC/MP Applications Cards;
September, page 9

SC/MP Board;
March, page 3

SC/MP Cross Software Packages;
by Gerry Madea, March, page 10

SC/MP Homebrew Computer System;
by Dan Grove, May, page 2

SC/MP Homebrew Computer System Additions;
by Dan Grove, July, page 11

SC/MP Keyboard Kit;
October, page 1

SC/MP Kit;
July, page 3

SC/MP Kit Replacement Parts and Repair Policy;
August, page 6

SC/MP Logic Symbol;
by Dan Grove, March, page 3

SCRIMP. . . with SC/MP;
November, page 6

Servicing of PACE Stack Overflow Interrupt;
by Subhash Bal, June, page 9

Simplified IMP-16C I/O Timing;
by Dan Grove, February, page 1

Subroutine Library Catalog;
March, page 3; September, page 8

Support Hardware for SC/MP and PACE Application Cards;
August, page 9

The 16-Bit Microprocessor;
by Phil Roybal, February, page 3

To All National Customers;
May, page 3

What is this Thing Called Relocatability;
by Ira McComic, January, page 2

Which Course Should You Attend;
August, page 5

Programs

IMP-16 Binary to BCD – SL002B BINBCD;
April, page 3

IMP-16 Data Transfer;
March, page 7

IMP-16 Disc RLM for PRMSFT B – SL0023A;
July, page 5

IMP-16 Disc RLM for PRMSFT C – SL0024A;
July, page 6

IMP-16 Insert Tabs – SL0026A TABTAP;
August, page 4

IMP-16 Message Routine to High-Speed Printer;
April, page 5

IMP-16 Paper Tape Titler – SL030A TITLER;
October, page 2

IMP-16 Print Plot & Test Plot – SL0016A PRTPLT &
SL0017A TSTPLT; February, page 5

IMP-16 RAM Dump – SL0012A RAMDUMP;
July, page 12

IMP-16 RLM Tape to Listing;
June, page 10

IMP-16 Serial I/O via UAR/T;
September, page 5

IMP-16 Square Root – SL0028A SQRT;
September, page 10

IMP-16 String Manipulation Package;
May, page 6

PACE Absolute LM Punch – SL0025A PALM;
August, page 3

PACE ASCII to Binary Conversion;
May, page 8

PACE Binary to ASCII – SL0022 NUMPRG;
June, page 5

PACE Binary to BCD;
October, page 7

PACE Calculator – SL0018A;
March, page 8

PACE, IMP-16 Insert Tabs – SL0026A TABTAP;
August, page 4

PACER Binary to BCD Conversion;
June, page 4

PACER Burn-in;
June, page 4

PACER Demos
1. Dice Simulator; March, page 5
2. Temperature Converter; March, page 5
3. Number Sorter; March, page6
4. Improved Dice Simulator; June, page 8
5. Roulette; June, page 8

SC/MP Complex Control Panel Driver;
May, page 3

SC/MP 6 x 8 Keyboard Matrix Scanner;
December, page 3

SC/MP Kit Terminal – SCRIMP;
Novermber, page 9

SC/MP Simple Control Panel Driver;
May, page 4

SC/MP Variable Delay;
June, page 6

Centerfolds

Data Bookshelf: Tools for the Design Engineer;
February

Functional analysis of National IMP microprocessor systems;
April

Functional analysis of National SC/MP microprocessor systems;
October

IMP-16 assembler programs, μspec 9;
January

INS8080A 8-Bit N-Channel Microprocessor, Data Sheet;
September

SC/MP Kit;
May

SC/MP Low-Cost Development System;
November

SC/MP Mates with Cassette Recorder, AN-163;
August

Supplement 3 to Pub. No. 4200021C;
July

Order Forms;
December

Technical Bulletin B76001;
July

Back issues of COMPUTE are available, while they last, at $1.00 for individual issues, or $5.00 for volume 1 and $10.00 for volume 2. Send your request and check to:

COMPUTE/208
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, CA. 95051

WRITE SOMETHING!

We are looking for articles on any of the following subjects:

Applications
Artificial intelligence
Block structured languages
Business data processing
Club news
Computer art
Debugging monitors
Hardware design
Homebrews
Interfacing
Kits
List processing
Microprocessors
Needs
Networks
New Products
Operating systems
Programming style
Programming techniques
Random Number Generators
Resources
Social implications of computers
Sorts
Technological trends
OR WHATEVER YOU ARE INTO

The contributions should be readable, and if it contains any software, please send along a source paper tape.

Now, what you will get for your effort, provided the article is accepted, is a complimentary membership to *COMPUTE*, the right to count coups over coffee, and a digital watch worth up to $100 retail (you specify what style, LED, LCD, mens, womens, gold, silver, etc.) or any of the nifty National calculators.

MICROPROCESSOR COURSES

	EASTERN TRAINING CENTER	WESTERN TRAINING CENTER
MICROPROCESSOR FUNDAMENTALS	June 6-9	June 6-9
SC/MP APPLICATIONS	June 13-16	June 20-23
PACE APPLICATIONS	June 20-23	June 13-16
ADVANCED PROGRAMMING	June 27-30	

TRAINING CENTER LOCATIONS

NATIONAL SEMICONDUCTOR
EASTERN MICROPROCESSOR TRAINING CENTER
1320 South Dixie Highway
Coral Gables, FL 33146
(305) 661-7969

NATIONAL SEMICONDUCTOR
WESTERN MICROPROCESSOR TRAINING CENTER
1333 Lawrence Expwy., Suite 430
Santa Clara, CA 95051
(408) 247-7924

ON THE ROAD WITH SC/MP

SC/MP Advanced Applications

Grand Rapids, MI	June 7-9	(616) 942-1320

SC/MP Applications

New Haven, CO	June 21-23	(203) 226-2527
Inglewood Cliff, NJ	June 21-23	(201) 461-5959

SC/MP WORKSHOPS IN EUROPE

DENMARK

June 20-22
Location: Copenhagen
Language: Danish
Contact: MULTIKOMPONENT A/S
Herstedvangen 7C
DK-2620 Albertslund
Telephone: 02-644477
Telex: 19155

ITALY

June 14-17
September 13-17
Location: Milano
Language: Italian
Contact: ADELSY S.P.A.
Via Domenicino 12
Milano 20749
Telephone: 02-4985051
Telex: 204339423

UNITED KINGDOM

Dates: Walk-in
Location: Berkshire
Language: English
Contact: N-SIGN
P.O. Box 119, Reading
Berkshire RG31NQ, England
Telephone: Reading (0734) 594911
Telex: 849391
Dates: On request
Location: Kent
Language: English
Contact: JERMYN INDUSTRIES
Vestry Estates, Sevenoaks
Kent, England
Telephone: Sevenoaks (0732) 50144
Telex: 95143

Contact your local National distributor, sales office, or Phil Hughes, National Semiconductor, Germany, for details of the SC/MP workshops and seminars.

Seventh Annual Institute in Computer Science

Announcing five intensive short courses to be given at the University's Santa Cruz campus, under the direction of Dr. William McKeeman, Professor of Information Science at UCSC, and an outstanding faculty.

Structured Programming Dr. Niklaus Wirth	JUL 11-12	$525
Data Base Management Dr. Michael R. Stonebraker and Dr. Eugene Wong	JUL 11-22	$525
Operating Systems Dr. Philip A. Bernstein	JUL 18-29	$525
Compiler Construction Dr. Franklin DeRemer	AUG 1-12	$525
Computer Graphics Dr. James Clark and Dr. Frank Crow	AUG 1-12	$525

For more information or for a complete brochure giving details of the courses, write to Joleen Kelsey, University of California Extension, Santa Cruz, CA 95064, or phone (408) 429-2671.

the Bit • Bucket

Gentlemen:

The Jefferson Amateur Radio Club and the Crescent City Computer Club would like to announce the New Orleans Hamfest/Computerfest will be held at the Hilton Inn in Kenner, LA (directly across from the New Orleans International Airport) September 24 & 25. This is the ARRL Delta Division Convention for 1977 and is the largest "Ham" outing in the deep south. This will not only be the largest Computerfest, but it is the *only* Computerfest in the area.

This year's event will feature a banquet Saturday night with entertainment, *two days* of commercial exhibits, fleamarkets and forums. There will also be a hospitality room, ladies events, FCC examinations and more.

This years grand prize is a complete Drake "C-Line" ham station and many door prizes will be awarded each day.

Information on tickets, room reservations and etc. will be furnished upon request by contacting the New Orleans Hamfest/Computerfest; P.O. Box 10111; Jefferson, LA 70181.

Thank you for your consideration.

Sincerely,

JAN R. EDWARDS
Publicity Chairman
P.O. Box 10111
Jefferson, LA 70181

Dear Georgia,

Thank you for sending me the NIBL listing. I have received it and here is the check for $15.00.

Since I have purchased a SC/MP LCDS am I automatically a member of COMPUTE? If I am would you please send me the program listings for:

SL0027A	SC/MP Math Package
SL0039A	SC/MP TAPEIO
SL0041A	SCSQRT

Sincerely,

Gary Rocky
ARE
P.O. Box 193
Carrollton, Texas 75006

Free COMPUTE membership forms are included with SC/MP Kits and LCDS systems. If the form is returned to National the person whose name appears on the form will become a member of COMPUTE. If you are already a member give the form to a friend. Free memberships are also given to all attendees of National's microprocessor courses.

Dear Ms. Marszalek:

I recently purchased a SC/MP and started receiving COMPUTE. I see references to a user library. Could I get a listing of the available programs?

Please send me a listing of the user library SL0027A SC/MP Math Package.

I would also like to receive a copy of AN-163 Mating SC/MP with a cassette recorder, as well as the corresponding object tape SL0039A. I have enclosed a check for $5.00

Is there any plan to update the ISP-8K/200, SC/MP Kit to use the new NMOS chip?

I have enjoyed COMPUTE. Thank you for your help.

Yours truly,

Gregory Constantine, Jr.
Lakeview Road
Poughkeepsie, New York 12603

1- SC/MP Math Package is listed in Appendix of SC/MP Applications Handbook ($5.00 from Marketing Services)
2- SC/MP Kit retrofit Kit is available from distributors (ISP-8K/205, $19.95). This includes a SC/MP-II, a new crystal, components and manual.

Dear Georgia:

We are the leading company in this field and our cross-assemblers are at least twenty times faster and more efficient than any others on the market. They also have a full macro and conditional assembly capability, providing considerable programming flexibility.

We will supply a cross-assembler for any commercially available microprocessor custom-tailored to a user's computer and his specific operating system. If his computer is made by Digital Equipment or Data General, the cross-assembler will be written in the assembly language of the host computer. Otherwise, it will be written in Fortran.

The price is $2,000 for the first cross-assembler purchased by a user, $1,500 for the second and $1,000 for all succeeding ones. If his first order is for two or more cross-assemblers, we will discount that order an additional $500.

We also have cross-assemblers for the 4040, 8080, PPS-8 and PACE available on several national time-sharing services. Cross-assemblers for the PPS-4, SC/MP and IMP-16 can, of course, also be used on time-sharing, but only if they are bought by the user at the above prices. In that case, there will be no payment of royalties for using the cross-assembler.

Should you have any further questions or need additional information, please don't hesitate to give me a call.

Sincerely,

Garret F. Ziffer
400 - 1 Totten Pond Road
Waltham, Massachusetts 02154
(617) 890-0888

Dear Georgia:

The first issue of COMPUTE which I received was Vol. 2, No. 7 dated July, 1976. On page 11 there was a small item by Dan Grove titled "SC/MP Homebrew Computer System Additions".

This obviously refers to an article in some issue previous to July, 1976. I wonder if it would be possible for me to obtain a copy of the issue that this article appeared in, or at least a xerox copy of the article in question.

Thank you,

Thomas M. Farr, Jr.
TMF Electronics
569 Medina Dr.
Lewisville, Texas 75067

The Homebrew System by D. Grove is in Vol. 2 #5 and some additions are in Vol. 2 #7. Back issues are available for $1.00 each from Compute until our supply is exhausted. This is reprinted in the April issue with some revisions.

Dear Sir:

Thank you for the 1975 Bit-Bucket Index published in the January COMPUTE.

Please send me the following back issues of COMPUTE: May, October, November, and December of 1975.

Very truly yours,
Brian H. Darley
Electronic Flight Controls

This issue contains the 1976 Bit Bucket/COMPUTE index.

LITERATURE REVIEW

IS-SW1—BUG BOOKS I AND II $17.00 (Set)

This comprehensive and well-illustrated combination text/workbook carries a user from basic concepts of digital electronics, such as gates and digital codes, to more sophisticated circuits that employ random access memories, sequencers, four-decade counters, and dot matrix displays. Ninety experiments demonstrate the characteristics of fifty different 7400-series integrated circuit chips. The text/workbook provides feedback to the user as he performs the experiments; reinforces the user's acquired knowledge with a series of questions at the end of each experiment.

IS-SW2—INSTRUCTOR'S MANUAL $3.00

This supplementary manual to the IS-SW1 is necessary for instructor use in the classroom and laboratory or for individual use in home-study situations. The Manual provides supplemental information including answers to the questions at the end of each experiment, suggestions for further reading, a discussion of the philosophy of the authors in their approach to digital electronics, and organizational suggestions for a lecture/laboratory course.

IS-SW4—BUGBOOK IIA $5.00

A supplement to Bugbooks I & II covering more advanced topics in Digital Electronics; in particular, transmission of data with asynchronous techniques using the ASC II code; how to use RS 232/20ma-current loops in combinations with the universal asynchronous receiver/transmitter.

IS-SW5—BUGBOOK III $15.00

This is unique text written in a self teaching style with experiments that thoroughly explain how to use, interface and program microcomputers. It focuses on a microprocessor system that uses the 8080 microprocessor and describes both in a general and specific manner. How to interface, how to program, how to work with the I/O busses, working with timing loops and more.

IS-SW8—MODULAR SELF-TAUGHT MICROPROCESSOR COURSE $30.00

The ideas of Bugbook III are presented in an entirely different format and aimed expressly at the individual interested in total self instruction. There is substantial amplification of certain parts of the course, particularly learning what programming is, different techniques/shortcuts, how to use it with the 8080 and in particular with the new MMD-1 Mini-Micro Designers, the basic microprocessor. The course aims at total comprehension directly reinforced by experiments.

The Bug Books may be ordered from:

IFM, Inc.
80 W. El Camino Real
Mountain View, CA 94040

Please include $1.50 for postage and handling.

New Reference Directory for Personal and Home Computing

A new, comprehensive reference directory for personal and home computing that will help every micro-computer enthusiast is being published by People's Computer Company. *PCC's Reference Book* brings together in one place listings of all possible sources of hardware, software, parts and services; of clubs, stores, periodicals, and books.

Another feature of *PCC's Reference Book* is the collection of in-depth articles on many varied aspects of the personal computing field.

PCC's Reference Book of Personal and Home Computing—Spring 1977 is available for $4.95 from most local computer stores or directly from People's Computer Company.

For further information contact:

Dwight McCabe
People's Computer Company
1010 Doyle St. #9
Box E
Menlo Park, CA 94025
(415) 323-3111

PACE Microprocessor

Low Cost Development System

The PACE Low Cost Development System from National Semiconductor is a fast and efficient vehicle for developing a working microprocessor application system. PACE LCDS features an ease of hardware and software access that facilitates interface and machine code development.

The system contains its own 20-key keypad. Using this, the designer can directly enter data, instructions, and control commands. And on the integral 6-digit hex LED display the designer can easily examine register and memory contents.

For versatility, the LCDS provides strap selectable baud rates and either RS232 or 20 mA current loop serial interfaces. These allow use of a variety of common input/output peripherals, including teletypewriters, CRT displays, and other keyboard devices.

The designer builds application routines or subroutines by entering code directly through a keyboard. Alternatively, he may use a teletypewriter to load paper tape in assembled format. Either way, he has complete flexibility in developing his programs. Using PACE LCDS, he may view, print, and modify memory and register contents. To simplify program checkout and debugging, both single-step and continuous mode with set breakpoints are available.

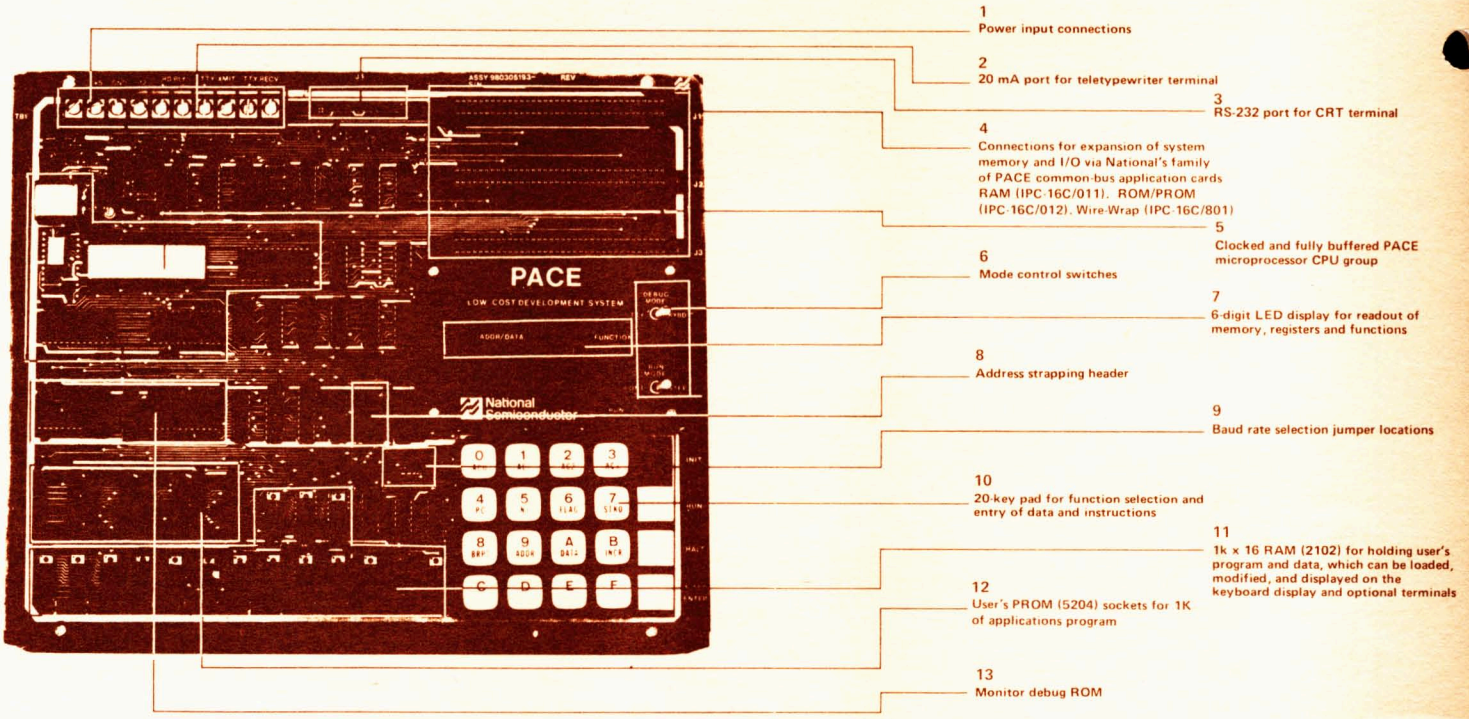
Once programs are debugged (LCDS contains a powerful debug monitor), the designer outputs the program on punched tape in a format compatible with National's PROM programmers.

Hardware interfaces may be quickly evaluated and debugged by plugging them into the motherboard and exercising them with the CPU. Three on-board connectors accept standard cards to further extend the RAM, ROM, and I/O capabilities. PACE LCDS features 60k x 16-bit addressability, with optional split base page operation.

The heart of PACE LCDS is the PACE CPU chip (IPC-16A/520D), supported by the System Timing Element (DP8302J) and Bidirectional Transceiver Elements (DP8300N). LCDS also includes 1K x 16 bits of random access memory, 1K x 16 bits of read-only memory, plus sockets for 1K x 16 bits of user-programmable read-only memory.

The PACE LCDS comes fully assembled and tested on a heavy duty printed circuit board mounted on a 10" x 12" x 3" aluminum chassis.

PACE LCDS: IPC-16P/301, Price: $585.


PACE
LOW COST DEVELOPMENT SYSTEM
National Semiconductor
1
Power input connections
2
20 mA port for teletypewriter terminal
3
RS-232 port for CRT terminal
4
Connections for expansion of system memory and I/O via National's family of PACE common-bus application cards RAM (IPC-16C/011), ROM/PROM (IPC-16C/012), Wire-Wrap (IPC-16C/801)
5
Clocked and fully buffered PACE microprocessor CPU group
6
Mode control switches
7
6-digit LED display for readout of memory, registers and functions
8
Address strapping header
9
Baud rate selection jumper locations
10
20-key pad for function selection and entry of data and instructions
11
1k x 16 RAM (2102) for holding user's program and data, which can be loaded, modified, and displayed on the keyboard display and optional terminals
12
User's PROM (5204) sockets for 1K of applications program
13
Monitor debug ROM

PACE DISK OPERATING SYSTEM

The PACE Disk Operating System is a powerful software development tool which operates as a peripheral to both IMP and PACE Microprocessor Development Systems.

PACE DOS has been architectured to current minicomputer software system standards, with features designed to reduce software development time and cost compared to other microprocessor disk-based development systems. Some of the PACE DOS features are:

Multiple Usage — with available cross-assemblers, PACE DOS can be used for software development for all NSC microprocessors.

Comprehensive File Management features include:

- Open-ended symbolically named files — by "naming" a file, the system automatically allocates/deallocates disk space, greatly improving disk utility and saving time.
- 4-level software file protection prevents inadvertent write-over.
- Utilities such as List File Directory, Copy File, Add/Delete/Rename/Append File, Copy Disk and Format Disk aid development efforts.

Macro Assembler — for easy development of high-level application oriented languages, allowing software architecting well in advance of final design decisions and simplifying software transfer between multiple types of processors. Features include:

- Wide range of listing controls including full or partial listing of macro expansion or conditional assemblies.
- Full range of conditional assembly directives including "if>, if<," and "if" character comparisons.
- Symbolically named or numbered macro parameters.
- Pool and pointer address automation.

Expanded Editor — Especially easy to use line oriented editor, including string search and substitution features — development money savers.

Full Utility Complement — File manager, assembler, editor, link editor (loader), and diagnostic are disk resident; debugger and boot loader are resident in firmware.
User Accessible RAM — The system runs in as little as 12K RAM, with up to 11K user accessible.

Dual Disk Drive — Soft-sectored IBM 3740 compatible, with high speed seeks.

Easy Implementation — Disk controller and firmware card, both included with system, plug directly into the development system backplane. The system also includes utility firmware, disk drives, power supply, housing, and all necessary cabling. Can be installed in IMP development systems using the IMP-PACE conversion kit.

Interested? Contact your local NSC representative or technical specialist for further information and data on how you can save time and money in software development.

PACE DOS: IPC-16P/840 Price: $4500

For further information call Bob Pecotich or Don Cooper (408) 737-6115 or contact your local National Semiconductor Office.

PACE DOS UPGRADE OR RETROFIT

Here are the typical configurations that Pace Imp-16 users might have and what they will need to upgrade to PACE DOS.

NOTE: A heavy duty pwr supply and min of 12K memory are basic requirements for PACE DOS.

1. IMP-16P/208 system with IMP DOS	PRICE
a. IPC-16P/100 PACE Conversion kit	$1000
b. Part # 930305457 PACE DOS ROM/PROM Card Kit, (includes IPC-16P/008B with 14 ea MM5204Q and IPC-16S/902M software on Diskettes	$1230
*c. IPC-16P/004A (as required for min of 12K memory)	$795
d. Part # 4004183 H.D. Pwr Supply (Z30X)	$550

2. IMP-16P/208 system only	
a. IPC-16P/100 PACE Conversion Kit	$1000
b. IPC-16P/840 PACE DOS	$4500
*c. IPC-16P/004A (as required for min of 12K memory)	$795
d. Part # 4004183 H.D. Pwr Supply (Z30X)	$550

3. IPC-16P/108 system only, to convert he needs:	
a. IPC-16P/840 PACE DOS	$1000
*b. IPC-16P/004A (as required for min of 12K memory)	$795
c. Part # 4004183 H.D. Pwr Supply (Z30X)	$550

Prototyping systems with backplane wiring (8302232) at Rev F and above can be done in the field. Rev E and below should be returned to the Service Center for retrofit.

*For those users whose systems have dynamic memory, the benefits are even greater if they convert to static memory. They will need:

a. IPC-16P/004A (as required for 12K)	$795
b. Part # 9802230 card cage chassis (Rev F or above wired for line printer, IMP-16C/400/500 cards and 16K of memory).	$600

To retain the dynamic memory requires:

a. IMP-16P/004 (as required for 12K dynamic memory card)	$770
b. Retrofit charge to update backplane for additional memory & DOS.	$550

For any assistance you may need for price, delivery, technical questions, entering orders, return of systems for retrofit, contact Microcomputer Marketing (408) 737-5546 or the Service Center (408) 737-6270.

(Continued from Page 1)

2. Add sockets and IC's as indicated:

LOCATION	SOCKET	I.C. TYPE	USE
3C	40 Pin	INS 8255	I/O Port
3D	14 Pin	74LS00	Address Decode
2D	16 Pin	16 Pin Header	I/O Connector
4D	16 Pin	16 Pin Header	I/O Connector

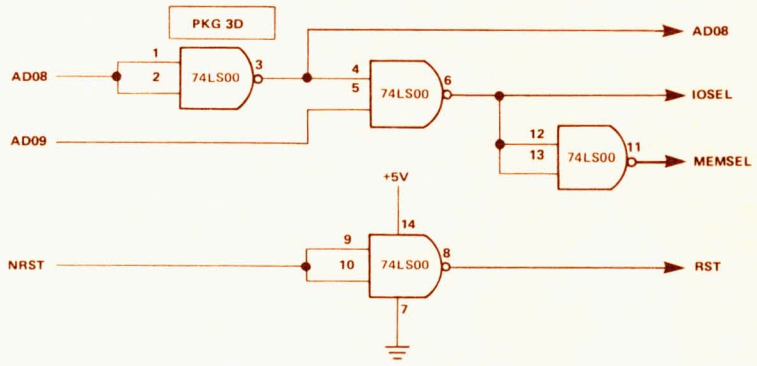
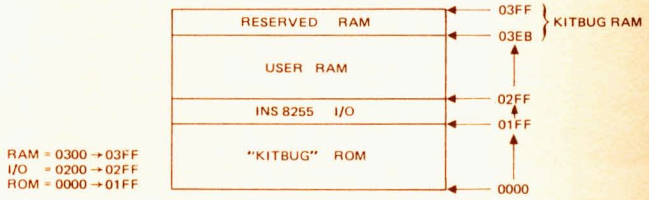


Figure 2

3. Wire the sockets as shown in figures 2 and 3, using the wire list in figure 4.

After making the changes, the following address map is used:



This pattern repeats itself throughout memory, because the address bits above AD09 are not decoded. Also, the following pattern repeats itself throughout the I/O area of memory:

0200 = I/O Port A, 0201 = I/O Port B
0202 = I/O Port C, 0203 = I/O Port Control Register

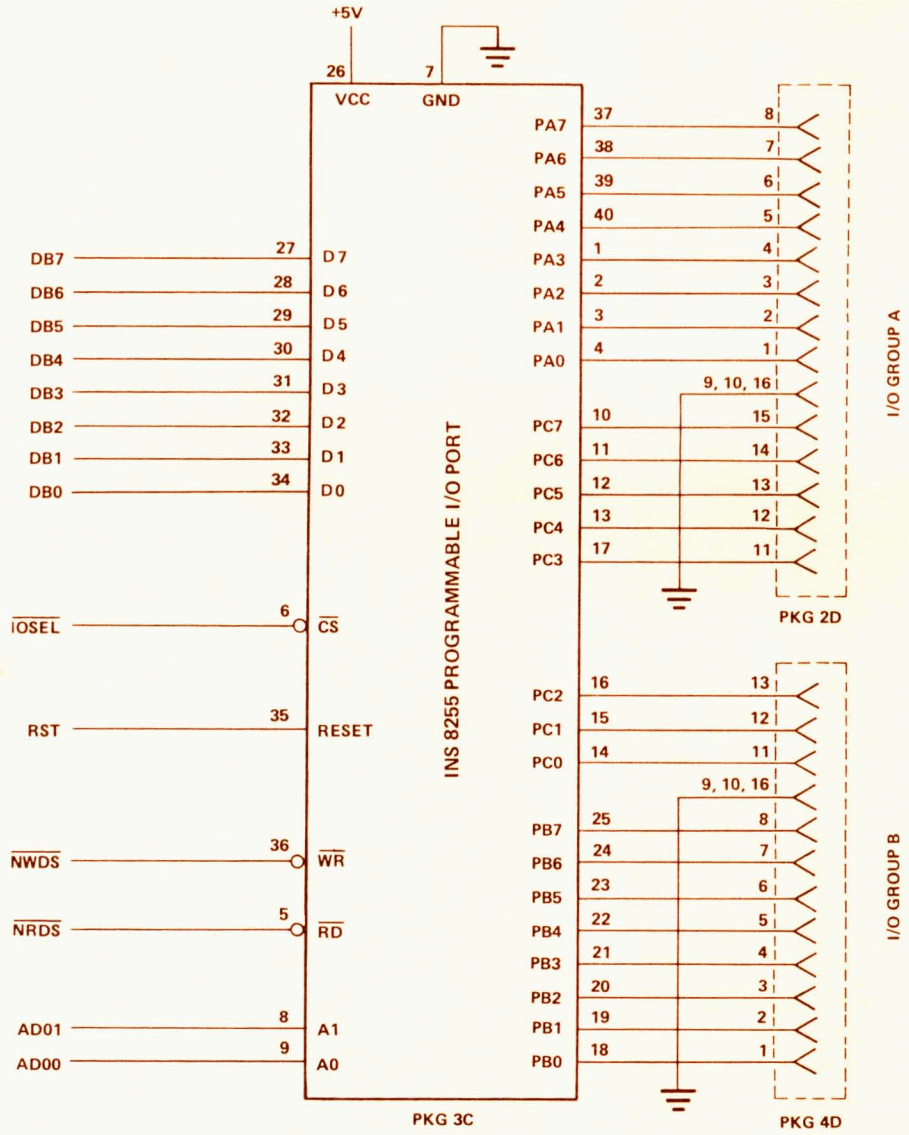
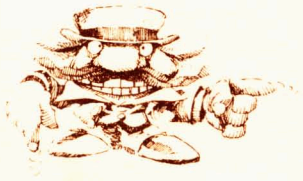


Figure 3

FROM PKG	FROM PIN	TO PKG	TO PIN
+5V		3C	26
+5V		3D	14
GND		3C	7
GND		3D	7
GND		2D	9
GND		4D	9
2D	9	2D	10
2D	10	2D	16
4D	9	4D	10
4D	10	4D	16
3D	3	3D	4
3D	3	3A	18 & 19
3A	18 & 19	4A	18 & 19
3D	1	3D	2
3D	1	3B	33
3D	5	3B	34
3D	6	3D	12
3D	6	3C	6
3D	12	3D	13
3D	11	J3	—
3D	9	3D	10
3D	9	3B	7
3D	8	3C	35
3C	5	3B	2
3C	8	3B	26
3C	9	3B	25
3C	36	3B	1
3C	1	2D	4
3C	2	2D	3
3C	3	2D	2
3C	4	2D	1
3C	10	2D	15
3C	11	2D	14
3C	12	2D	13
3C	13	2D	12
3C	14	4D	11
3C	15	4D	12
3C	16	4D	13
3C	17	2D	11
3C	18	4D	1
3C	19	4D	2
3C	20	4D	3
3C	21	4D	4
3C	22	4D	5
3C	23	4D	6
3C	24	4D	7
3C	25	4D	8
3C	37	2D	8
3C	38	2D	7
3C	39	2D	6
3C	40	2D	5

Figure 4



SC/MP Debugging Aid

Below is a handy reference chart for SC/MP programmers handcoding or debugging. Our thanks to:

J. B. Ross, Physics Dept., Park College, Kansas City, MO 64152

SC/MP MICROPROCESSOR INSTRUCTION SUMMARY

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	HALT	XAE	CCL	SCL	DINT	IEN	CSA	CAS	NOP	X	X	X	X	X	X	X
1	X	X	X	X	X	X	X	X	X	SI0	X	X	SR	SRL	RR	RRL
2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3	XPAL0	XPAL1	XPAL2	XPAL3	XPAH0	XPAH1	XPAH2	XPAH3	X	X	X	X	XPPC0	XPPC1	XPPC2	XPPC3
4	LDE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
5	ANE	X	X	X	X	X	X	X	ORE	X	X	X	X	X	X	X
6	XRE	X	X	X	X	X	X	X	DAE	X	X	X	X	X	X	X
7	ADE	X	X	X	X	X	X	X	CAE	X	X	X	X	X	X	X
8	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	DLY*
9	JMP0*	JMP1*	JMP2*	JMP3*	JP0*	JP1*	JP2*	JP3*	JZ0*	JZ1*	JZ2*	JZ3*	JNZ0*	JNZ1*	JNZ2*	JNZ3*
A	X	X	X	X	X	X	X	X	ILD0*	ILD1*	ILD2*	ILD3*	X	X	X	X
B	X	X	X	X	X	X	X	X	DLD0*	DLD1*	DLD2*	DLD3*	X	X	X	X
C	LD0*	LD1*	LD2*	LD3*	LDI*	LD@1*	LD@2*	LD@3*	ST0*	ST1*	ST2*	ST3*	X	ST@1*	ST@2*	ST@3*
C	AND0*	AND1*	AND2*	AND3*	ANI*	AND@1*	AND@2*	AND@3*	OR0*	OR1*	OR2*	OR3*	ORI*	OR@1*	OR@2*	OR@3*
E	XOR0*	XOR1*	XOR2*	XOR3*	XRI*	XOR@1*	XOR@2*	XOR@3*	DAD0*	DAD1*	DAD2*	DAD3*	DAI*	DAD@1*	DAD@2*	DAD@3*
F	ADD0*	ADD1*	ADD2*	ADD3*	ADI*	ADD@1*	ADD@2*	ADD@3*	CAD0*	CAD1*	CAD2*	CAD3*	CAI*	CAD@1*	CAD@2*	CAD@3*

notes: X indicates unimplemented code

* indicates a two byte instruction

SC/MP-II Retrofit Kit

The success of SC/MP Kit and SC/MP Keyboard Kit has been so overwhelming that it has inspired a brand new kit for SC/MP-II.

It is called the "SC/MP-II Retrofit Kit". The SC/MP-II Retrofit Kit enables SC/MP Kit, SC/MP Keyboard Kit, and potential SC/MP Users to evaluate SC/MP-II Microprocessor object code programming and pinout compatibilities — at very low cost within a few minutes.

The SC/MP-II Retrofit Kit includes:

- SC/MP-II CPU Chip
- 2 MHz Crystal for Timing
- Resistors, Capacitors, Wires, Etc.
- Application/Users Manual, Data Sheet, Etc.

The Order Number and suggested retail price for one or more SC/MP-II Retrofit Kit is as follows:

	1-24
ISP-8K/205	$18.50

SC/MP-II RETROFIT KIT FIRMWARE CHANGES

If you have retrofitted your SC/MP p-channel kit to a SC/MP n-channel, you may wish to change the delays in the TTY routines to reflect the increased speed of SC/MP-II. The following are the firmware changes for Kit Bug (reference: appendix B, SC/MP Kit User's Manual)

line #	address	object code	n-channel instruction
338	018F	C4C3	LDI 0C3
339	0191	8F04	DLY 8
346	019C	C445	LDI 045
347	019E	8F11	DLY 011
366	01BB	8F11	DLY 011
381	01C6	C4BB	LDI 0BB
382	01C8	8F2F	DLY 02F
388	01D2	C454	LDI 054
389	01D4	8F11	DLY 011

continuing education

Cogswell College, an accredited, private, non-profit technical institution is offering upper-division courses to be credited towards the Bachelor of Science in Engineering Technology Degree given by Cogswell College. Evening courses in microprocessors for the Summer quarter are:

ET452 Microprocessors and Applications
3 units 5:30-8:30 pm
Prerequisite: Familiarity with MOS. I.C.S. Basic microprocessor hardware organization and operation.

ET453 Microcomputer Programming
3 units 5:30-8:30 pm
Prerequisite: ET452 or familiarity with microprocessor hardware. Programming languages and software techniques of the 4040 and 8080 systems to be discussed in detail.

Tuition is $45 per unit
Scheduling begins on *June 27, 1977*

For further information contact:
Cogswell College Santa Clara Valley
Education Center at Fairchild
441 Whisman Rd.
Mountain View, Ca., 94042 (415) 962-3815
San Francisco Office (415) 433-1994

UNITED STATES
COMPUTE/208
National Semiconductor Corp.
2900 Semiconductor Drive
Santa Clara, CA. 95051
Tel: (408) 247-7924
TWX: 910-338-0537

GERMANY
National Semiconductor GmbH
808 Fuerstenfeldbruck
Industriestrasse 10
Tel: 0814/1371
Telex: 05-27649

AUSTRALIA
NS Electronics Pty Ltd.
Cnr. Stud Road & Mtn. Highway
Bayswater, Victoria 3153
Tel: 03-729-6333
Telex: 32096